

DOCUMENT PROCESSING METHOD AND SYSTEM THEREFOR

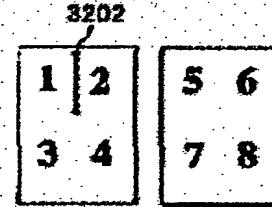
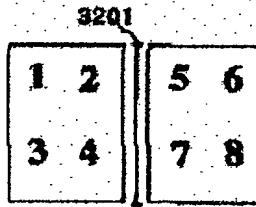
Patent number: JP2003091518
Publication date: 2003-03-28
Inventor: MORI YASUO
Applicant: CANON INC
Classification:
 - International: G06F17/21; G06F3/00
 - European:
Application number: JP20010280756 20010914
Priority number(s):

Also published as:
 US2003056180 (A1)

Abstract of JP2003091518

PROBLEM TO BE SOLVED: To provide a document processing method and a system therefore implementing display improving usability and efficiency of editing operation by use of hierarchical holding of data and a set value when inserting, moving and copying/pasting data in the document processing system hierarchically holding the data and the set value.

SOLUTION: When dragging a figure showing a desired document and moving the figure on documents in the case of movement or a copy of the desired document data to a prescribed position of the documents in a document process editing the documents comprising the plurality of documents data, a boundary of the document data in the documents nearest to a position of a cursor dragging the figure showing the desired document is detected, and a distinct indicator is displayed in the boundary of the document data of the documents.



(19)日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特願2003-91518

(P2003-91518A)

(43)公開日 平成15年3月28日 (2003.3.28)

(51)Int.Cl. ⁷	識別記号	F I	データコード(参考)
G 0 6 F 17/21	5 0 1	G 0 6 F 17/21	5 0 1 Z 5 B 0 0 9
	5 7 0		5 7 0 N 5 E 5 0 1
	5 8 6		5 8 6 B
3/00	6 6 1	3/00	6 5 1 B

審査請求 未請求 請求項の数27 O.L (全33頁)

(21)出願番号	特願2001-280758(P2001-280758)	(71)出願人	000001007 キヤノン株式会社 東京都大田区下丸子3丁目30番2号
(22)出願日	平成13年9月14日 (2001.9.14)	(72)発明者	森 安生 東京都大田区下丸子3丁目30番2号 キヤ ノン株式会社内

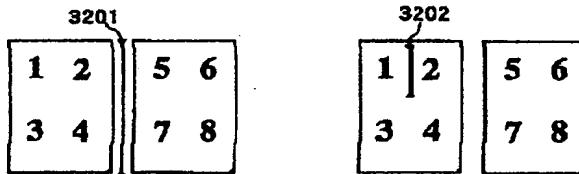
(74)代理人	100076428 弁理士 大塚 康徳 (外3名) F ターム (参考) 5B009 RB18 SA05 SA12 TB04 5E501 AA14 AC34 BA05 CA02 EA07 EB12 EB13 EB14 EB15 FA13 FA14 FB22
---------	---

(54)【発明の名称】 文書処理方法及びそのシステム

(57)【要約】

【課題】 本発明では、データと設定値を階層的に保持するような文書処理システムで、データの挿入・移動・コピペーストする際に、階層的にデータと設定値を保持しているという特徴を最大限に活かして、編集操作の効率化とユーザビリティを向上させる表示を実現する文書処理方法及びそのシステムを提供する。

【解決手段】 複数の原稿データからなる文書を編集する文書処理で、所望の原稿データを文書の所定位置へ移動又は複製する場合に、前記所望の原稿を表す図形をドラッグして、文書上を移動させると、前記所望の原稿を表す図形をドラッグするカーソルの位置に最も近い、文書中の原稿データの境界を検出し、該文書中の原稿データの境界に、識別可能な標識を表示する。



【特許請求の範囲】

【請求項1】 複数の原稿データからなる文書を編集する文書処理方法であって、

所望の原稿データを文書の所定位置へ移動又は複製する場合に、前記所望の原稿を表す図形をドラッグして、文書上を移動させると、前記所望の原稿を表す図形をドラッグするカーソルの位置に最も近い、文書中の原稿データの境界を検出し、該文書中の原稿データの境界に、識別可能な標識を表示することを特徴とする文書処理方法。

【請求項2】 前記識別可能な標識は前記文書中の原稿データの境界を示すバーであることを特徴とする請求項1記載の文書処理方法。

【請求項3】 前記バーを、他の図形よりも太い、濃い、あるいは異なる色で表示することを特徴とする請求項2記載の文書処理方法。

【請求項4】 前記文書が原稿データのツリー構造で記憶されて、ツリー構造が見える文書構造表示画面とツリー構造が見えない文書構造表示画面とが表示可能であり、

該ツリー構造が見えない文書構造表示画面においては、階層の高い境界にノードの所属が分かるバーを割り当てるることを特徴とする請求項1記載の文書処理方法。

【請求項5】 ツリー構造が見える文書構造表示領域とツリー構造が見えない文書構造表示領域とが、1つの画面に表示され、

カーソルの位置がツリー構造が見えない文書構造表示画面領域にある場合に、階層の高い境界にノードの所属が分かるバーを割り当てるることを特徴とする請求項1記載の文書処理方法。

【請求項6】 前記ツリー構造が見える文書構造表示は文書の構成をツリー形式で表現するツリー表示であり、前記ツリー構造が見えない文書構造表示は印刷データの内容を表示するプレビュー表示であることを特徴とする請求項4又は5記載の文書処理方法。

【請求項7】 前記ノードの所属が分かる境界のバーは2本のバーからなり、有効な1本を表示することを特徴とする請求項4又は5記載の文書処理方法。

【請求項8】 前記ノードの所属が分かる境界のバーは2本のバーからなり、有効な1本を他方と識別可能に表示することを特徴とする請求項4又は5記載の文書処理方法。

【請求項9】 前記ツリー構造は文書、章、ページの階層構造からなることを特徴とする請求項4乃至8のいずれか1つに記載の文書処理方法。

【請求項10】 カーソルの位置が、複数の原稿データが1ページの印刷データに合成されていることを表示する文書構造表示画面にある場合に、

前記カーソルに最も近い原稿データの境界が印刷データの境界であるか否かを判定し、

印刷データの境界であるか否かを識別可能な異なるバーを割り当てるなどを特徴とする請求項1記載の文書処理方法。

【請求項11】 前記ドラッグしていた原稿データをドロップすると、前記標識が表示された位置に前記ドラッグしていた原稿データを挿入することを特徴とする請求項1記載の文書処理方法。

【請求項12】 複数の原稿データからなる文書を編集する文書処理方法であって、前記文書がツリー構造を有し、

前記原稿データ及びノードの各々が属性を有し、該属性が原稿データ及びノードに固有であるか否かを識別可能な標識を、原稿データ及びノードの図形と共に表示することを特徴とする文書処理方法。

【請求項13】 複数の原稿データからなる文書を編集する文書処理システムであって、

所望の原稿データを文書の所定位置へ移動又は複製する場合に、前記所望の原稿を表す図形をドラッグして、文書上を移動させると、

前記所望の原稿を表す図形をドラッグするカーソルの位置に最も近い、文書中の原稿データの境界を検出する境界検出手段と、

該文書中の原稿データの境界に、識別可能な標識を表示する標識表示手段とを有することを特徴とする文書処理システム。

【請求項14】 前記識別可能な標識は前記文書中の原稿データの境界を示すバーであることを特徴とする請求項13記載の文書処理システム。

【請求項15】 前記標識表示手段は、前記バーを他の図形よりも太い、濃い、あるいは異なる色で表示することを特徴とする請求項14記載の文書処理システム。

【請求項16】 前記文書処理システムは、前記文書を原稿データのツリー構造で記憶し、ツリー構造が見える文書構造表示画面とツリー構造が見えない文書構造表示画面とが表示可能であり、

前記標識表示手段は、該ツリー構造が見えない文書構造表示画面においては、階層の高い境界にノードの所属が分かるバーを割り当てるなどを特徴とする請求項13記載の文書処理システム。

【請求項17】 前記文書処理システムは、ツリー構造が見える文書構造表示領域とツリー構造が見えない文書構造表示領域とを、1つの画面に表示し、

前記標識表示手段は、カーソルの位置がツリー構造が見えない文書構造表示領域にある場合に、階層の高い境界にノードの所属が分かるバーを割り当てるなどを特徴とする請求項13記載の文書処理システム。

【請求項18】 前記ツリー構造が見える文書構造表示は文書の構成をツリー形式で表現するツリー表示であり、前記ツリー構造が見えない文書構造表示は印刷データの内容を表示するプレビュー表示であることを特徴と

する請求項16又は17記載の文書処理システム。

【請求項19】前記ノードの所属が分かれる境界のバーは2本のバーからなり、前記標識表示手段は、有効な1本を表示することを特徴とする請求項16又は17記載の文書処理システム。

【請求項20】前記ノードの所属が分かれる境界のバーは2本のバーからなり、前記標識表示手段は、有効な1本が他方と識別可能に表示することを特徴とする請求項16又は17記載の文書処理システム。

【請求項21】前記ツリー構造は文書、章、ページの階層構造からなることを特徴とする請求項13乃至20のいずれか1つに記載の文書処理システム。

【請求項22】カーソルの位置が、複数の原稿データが1ページの印刷データに合成されていることを表示する文書構造表示画面にある場合に、

前記カーソルに最も近い原稿データの境界が印刷データの境界であるか否かを判定する判定手段を更に有し、

前記標識表示手段は、印刷データの境界であるか否かを識別可能な異なるバーを割り当てるなどを特徴とする請求項13記載の文書処理システム。

【請求項23】前記ドラッグしていた原稿データをドロップすると、前記標識が表示された位置に前記ドラッグしていた原稿データを挿入する原稿データ挿入手段を更に有することを特徴とする請求項13記載の文書処理システム。

【請求項24】複数の原稿データからなる文書を編集する文書処理システムであって、

前記文書がツリー構造を有し、

前記原稿データ及びノードの各々が属性を有し、該属性が原稿データ及びノードに固有であるか否かを判別する判別手段と、

該判別結果が識別可能な標識を原稿データ及びノードの図形と共に表示する表示手段とを有することを特徴とする文書処理システム。

【請求項25】複数の原稿データからなる文書を編集する文書処理プログラムであって、

所望の原稿データを文書の所定位置へ移動又は複製する場合に、前記所望の原稿を表す図形をドラッグして、文書上を移動させると、

前記所望の原稿を表す図形をドラッグするカーソルの位置に最も近い、文書中の原稿データの境界を検出する境界検出モジュールと、

該文書中の原稿データの境界に、識別可能な標識を表示する標識表示モジュールとを含むことを特徴とする文書処理プログラム。

【請求項26】ツリー構造が見える文書の構成をツリーフォームで表現するツリー表示領域とツリー構造が見えない印刷データの内容を表示するプレビュー表示領域とが、1つの画面に表示され、

カーソルの位置が前記いずれの領域にあるかを判別する

判別モジュールを更に含み、

前記標識表示モジュールは、プレビュー表示領域にある場合に、階層の高い境界にノードの所属が分かれる標識を割り当てるなどを特徴とする請求項25記載の文書処理プログラム。

【請求項27】複数の原稿データからなる文書を編集する文書処理プログラムであって、

前記文書がツリー構造を有し、前記原稿データ及びノードの各々が属性を有し、該属性が原稿データ及びノードに固有であるか否かを識別可能な標識を、原稿データ及びノードの図形と共に表示する標識表示モジュールを含むことを特徴とする文書処理プログラム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は文書処理方法及びそのシステム、例えば文書処理プログラムや画像編集プログラム等、様々なプログラムによって生成される出力データをひとつの文書としてまとめ、その文書に対する編集機能を提供する文書処理方法及びそのシステムに関し、特にパーソナルコンピュータ等の情報処理とプリンタから成るシステムにおける文書処理方法及びそのシステムに関するものである。

【0002】

【従来の技術】文字や表、画像など、データの種類が異なると、データを定義する構造やそれらデータに対する編集操作が異なるために、データの種類に応じた様々なアプリケーションプログラムが提供されている。利用者は、文字を編集するためには文字処理プログラム、表を編集するためには表計算プログラム、画像を編集するためには画像編集プログラムといった具合に、データの種類別にアプリケーションを使い分けている。

【0003】このように、利用者はデータの種類ごとにアプリケーションプログラムを使い分けるのが一般的である。しかし、利用者が作成しようとする文書としては、例えば文字のみや、表のみ、画像のみといった1種類のデータのみから構成される文書よりも、文字と表、文字と画像など、複数種類のデータから構成される文書の方が一般的である。そこで複数種類のデータを含む目的文書を作成するために、利用者は各種アプリケーションが備えている印刷機能を利用してアプリケーションごとにデータを印刷させ、印刷物を所望の順序に組み合わせる必要があった。

【0004】あるいは、いわゆるオフィススイートと呼ばれる、各種アプリケーションによってひとつの統合アプリケーションを形成するプログラムには、各アプリケーションで生成されたデータを組み合わせ、ひとつの文書を構成する機能を提供するものもある。この統合アプリケーションを使用する場合には、利用者は、各アプリケーションで作成したデータを、統合アプリケーション

に含まれる特定のアプリケーションによってひとつの目的文書にまとめることができる。

【0005】しかしながら、利用者が各種アプリケーションによる印刷物を組み合わせてひとつの目的文書を作成する場合、例えばページ番号を各ページに振るために、いったん必要なデータをすべて印刷出力し、文書としてまとめてから付すべきページ番号を決定する必要がある。そして、各アプリケーションによって、そのアプリケーションで作成された原稿の各ページ（これを論理ページあるいは原稿ページと呼ぶ）ごとに、決定されたページ番号を書き込んでいく。アプリケーションプログラムがページ番号を振る機能を有している場合であっても、不連続部分があればその部分についてはやはり利用者がページ番号を指定しなければならない。また、目的文書のページを再配置した場合には、ページ番号もそれに合わせて振り直さなければならない。あるいは、複数の原稿ページを、印刷物としての1ページ（これを物理ページあるいは印刷ページと呼ぶ）にまとめたり、片面印刷から両面印刷に変更するなど、データの内容にかかる変更ではなく、単に書式を変更する場合であっても、アプリケーションによって編集及び印刷のし直しが必要となる。

【0006】このように、データの種類ごとにそれを管理できるアプリケーションが異なることから、アプリケーション間のインターフェースを利用者自身が人手によって提供しなければならない。このことは、利用者に多大な労力が要求されるということであり、生産性の低下をもたらす。さらに、多くの人手を介することからエラーの発生も生じやすかった。

【0007】一方、統合アプリケーションを利用して目的文書を作成する場合、印刷出力せずに、データの状態で各種データを配置することができる。このため、印刷物を組み合わせて目的文書を作成する場合ほどには労力は必要とされることはない。しかしながら、各種データを編集・作成するためのアプリケーションは、統合アプリケーションに含まれるものに限られてしまい、利用者が望むアプリケーションが使用できるとは限らない。また、統合アプリケーションによって作成された目的文書はひとつの文書ファイルであり、ファイル単位で編集・出力等の管理が行われる。そのために、文書ファイルの一部について書式を設定しようとしてもアプリケーションの機能による制約が多く、例えば書式が変わる部分ごとに書式設定を変更しては印刷し直すといった作業が必要とされる。このため、多くの労力が必要とされて生産性が低い点においては、前述した方法と変わりはなかった。

【0008】

【発明が解決しようとする課題】本出願人は、先にデータをページ単位で保持し、各種設定値にしたがって、用紙上にレイアウトして印刷するようなシステムを提案し

た。しかしながら、このようなシステムでは、データを階層構造で保持していても、設定値は文書全体でしか保持しておらず、部分的な設定をすることはできなかつた。例えば、代表的なレイアウト設定であるNページ印刷を例に挙げると、文書全体で2ページ印刷の設定はできるが、そのうちの一部分を4ページ印刷に変更することはできなかつた。また、部分毎に設定ができるようなシステムでは、各部分だけで属性を保持しているために、全体を統一して一齊に変更することができなかつた。そこで、データを階層的に保持するだけでなく、各種設定項目も各階層に保持することで、全体を統一することも、部分的な設定をすることも可能であるようなシステムが望まれる。

【0009】このようなシステムでは、編集操作として、設定値の変更だけでなく、ページ単位やページをまとめた階層など任意の部分木を移動したりコピーベーストしたりすることが可能となる。

【0010】そこで本発明では、データと設定値を階層的に保持するような文書処理システムで、データの挿入・移動・コピーベーストする際に、階層的にデータと設定値を保持しているという特徴を最大限に活かして、編集操作の効率化とユーザビリティを向上させる表示を実現する文書処理方法及びそのシステムを提供することを目的とする。

【0011】

【課題を解決するための手段】この課題を解決するために、本発明の文書処理方法は、複数の原稿データからなる文書を編集する文書処理方法であって、所望の原稿データを文書の所定位置へ移動又は複製する場合に、前記所望の原稿を表す図形をドラッグして、文書上を移動させると、前記所望の原稿を表す図形をドラッグするカーソルの位置に最も近い、文書中の原稿データの境界を検出し、該文書中の原稿データの境界に、識別可能な標識を表示することを特徴とする。

【0012】ここで、前記識別可能な標識は前記文書中の原稿データの境界を示すバーである。また、前記バーを、他の図形よりも太い、濃い、あるいは異なる色で表示する。また、前記文書が原稿データのツリー構造で記憶されて、ツリー構造が見える文書構造表示画面とツリー構造が見えない文書構造表示画面とが表示可能であり、該ツリー構造が見えない文書構造表示画面においては、階層の高い境界にノードの所属が分かるバーを割り当てる。また、ツリー構造が見える文書構造表示領域とツリー構造が見えない文書構造表示領域とが、1つの画面に表示され、カーソルの位置がツリー構造が見えない文書構造表示領域にある場合に、階層の高い境界にノードの所属が分かるバーを割り当てる。また、前記ツリー構造が見える文書構造表示は文書の構成をツリー形式で表現するツリー表示であり、前記ツリー構造が見えない文書構造表示は印刷データの内容を表示するプレビ

ユー表示である。また、前記ノードの所属が分かれる境界のバーは2本のバーからなり、有効な1本を表示する。また、前記ノードの所属が分かれる境界のバーは2本のバーからなり、有効な1本を他方と識別可能に表示する。また、前記ツリー構造は文書、章、ページの階層構造からなる。また、カーソルの位置が、複数の原稿データが1ページの印刷データに合成されていることを表示する文書構造表示画面にある場合に、前記カーソルに最も近い原稿データの境界が印刷データの境界であるか否かを判定し、印刷データの境界であるか否かを識別可能な異なるバーを割り当てる。また、前記ドラッグしていた原稿データをドロップすると、前記標識が表示された位置に前記ドラッグしていた原稿データを挿入する。

【0013】又、本発明の文書処理方法は、複数の原稿データからなる文書を編集する文書処理方法であって、前記文書がツリー構造を有し、前記原稿データ及びノードの各々が属性を有し、該属性が原稿データ及びノードに固有であるか否かを識別可能な標識を、原稿データ及びノードの図形と共に表示することを特徴とする。

【0014】又、本発明の文書処理システムは、複数の原稿データからなる文書を編集する文書処理システムであって、所望の原稿データを文書の所定位置へ移動又は複製する場合に、前記所望の原稿を表す図形をドラッグして、文書上を移動させると、前記所望の原稿を表す図形をドラッグするカーソルの位置に最も近い、文書中の原稿データの境界を検出する境界検出手段と、該文書中の原稿データの境界に、識別可能な標識を表示する標識表示手段とを有することを特徴とする。

【0015】ここで、前記識別可能な標識は前記文書中の原稿データの境界を示すバーである。また、前記標識表示手段は、前記バーを他の図形よりも太い、濃い、あるいは異なる色で表示する。また、前記文書処理システムは、前記文書を原稿データのツリー構造で記憶し、ツリー構造が見える文書構造表示画面とツリー構造が見えない文書構造表示画面とが表示可能であり、前記標識表示手段は、該ツリー構造が見えない文書構造表示画面においては、階層の高い境界にノードの所属が分かれるバーを割り当てる。また、前記文書処理システムは、ツリー構造が見える文書構造表示領域とツリー構造が見えない文書構造表示領域とを、1つの画面に表示し、前記標識表示手段は、カーソルの位置がツリー構造が見えない文書構造表示領域にある場合に、階層の高い境界にノードの所属が分かれるバーを割り当てる。また、前記ツリー構造が見える文書構造表示は文書の構成をツリー形式で表現するツリー表示であり、前記ツリー構造が見えない文書構造表示は印刷データの内容を表示するプレビュー表示である。また、前記ノードの所属が分かれる境界のバーは2本のバーからなり、前記標識表示手段は、有効な1本を表示する。また、前記ノードの所属が分かれる境界のバーは2本のバーからなり、前記標識表示手段は、

有効な1本が他方と識別可能に表示する。また、前記ツリー構造は文書、章、ページの階層構造からなる。また、カーソルの位置が、複数の原稿データが1ページの印刷データに合成されていることを表示する文書構造表示画面にある場合に、前記カーソルに最も近い原稿データの境界が印刷データの境界であるか否かを判定する判定手段を更に有し、前記標識表示手段は、印刷データの境界であるか否かを識別可能な異なるバーを割り当てる。また、前記ドラッグしていた原稿データをドロップすると、前記標識が表示された位置に前記ドラッグしていた原稿データを挿入する原稿データ挿入手段を更に有する。

【0016】又、本発明の文書処理システムは、複数の原稿データからなる文書を編集する文書処理システムであって、前記文書がツリー構造を有し、前記原稿データ及びノードの各々が属性を有し、該属性が原稿データ及びノードに固有であるか否かを判別する判別手段と、該判別結果が識別可能な標識を原稿データ及びノードの図形と共に表示する表示手段とを有することを特徴とする。

【0017】又、本発明の文書処理プログラムは、複数の原稿データからなる文書を編集する文書処理プログラムであって、所望の原稿データを文書の所定位置へ移動又は複製する場合に、前記所望の原稿を表す図形をドラッグして、文書上を移動させると、前記所望の原稿を表す図形をドラッグするカーソルの位置に最も近い、文書中の原稿データの境界を検出する境界検出手段と、該文書中の原稿データの境界に、識別可能な標識を表示する標識表示モジュールとを含むことを特徴とする。ここで、ツリー構造が見える文書の構成をツリー形式で表現するツリー表示領域とツリー構造が見えない印刷データの内容を表示するプレビュー表示領域とが、1つの画面に表示され、カーソルの位置が前記いずれの領域にあるかを判別する判別モジュールを更に含み、前記標識表示モジュールは、プレビュー表示領域にある場合に、階層の高い境界にノードの所属が分かれる標識を割り当てる。

【0018】又、本発明の文書処理プログラムは、複数の原稿データからなる文書を編集する文書処理プログラムであって、前記文書がツリー構造を有し、前記原稿データ及びノードの各々が属性を有し、該属性が原稿データ及びノードに固有であるか否かを識別可能な標識を、原稿データ及びノードの図形と共に表示する標識表示モジュールを含むことを特徴とする。

【0019】

【発明の実施の形態】以下、添付図面に従って、本発明の実施形態を詳細に説明する。

【0020】【本実施形態の文書処理システムの概要】本発明の第1の実施形態である文書処理システムの概要を、図1～図13を参照して説明する。この文書処理シ

ステムでは、一般アプリケーションにより作成されたデータファイルが、電子原稿ライタによって電子原稿ファイルに変換される。製本アプリケーションはその電子原稿ファイルを編集する機能を提供している。尚、本例では、それぞれの機能が明瞭になるように、一般アプリケーション、電子原稿ライタ、製本アプリケーション、電子原稿デスプーラと分離して示しているが、ユーザに提供されるパッケージはこれらに限定されず、これらを組合せたアプリケーションやグラフィックエンジンとして提供されてもよい。以下、その詳細は説明する。

【0021】<本実施形態の文書処理システムのソフトウェア構成例>図1は、本実施形態の文書処理システムのソフトウェア構成を示す図である。文書処理システムは、本発明の文書処理装置の好適な実施形態であるデジタルコンピュータ100（以下、ホストコンピュータとも呼ばれる）によって実現されている。一般アプリケーション101は、ワードプロセシングやスプレッドシート、フォトレタッチ、ドローあるいはペイント、プレゼンテーション、テキスト編集などの機能を提供するアプリケーションプログラムであり、OSに対する印刷機能を有している。これらアプリケーションは、作成された文書データや画像データなどのアプリケーションデータを印刷するにあたって、オペレーティングシステム（OS）により提供される所定のインタフェース（一般に、GDIと呼ばれる）を利用する。すなわち、アプリケーション101は、作成したデータを印刷するために、前記インタフェースを提供するOSの出力モジュールに対して、あらかじめ定められる、OSに依存する形式の出力コマンド（GDI関数と呼ばれる）を送信する。出力コマンドを受けた出力モジュールは、プリンタ等の出力デバイスが処理可能な形式にそのコマンドを変換し、変換されたコマンド（DDI関数と呼ばれる）を出力する。出力デバイスが処理可能な形式はデバイスの種類やメーカー、機種などによって異なるために、デバイスごとにデバイスドライバが提供されており、OSではそのデバイスドライバを利用してコマンドの変換を行い、印刷データを生成し、JL（Job Language）でくくることにより印刷ジョブが生成される。OSとしてマイクロソフト社のウインドウズを利用する場合には、前述した出力モジュールとしてはGDI（Graphic Device Interface）と呼ばれるモジュールが相当する。

【0022】電子原稿ライタ102は、前述のデバイスドライバを改良したものであり、本文書処理システム実現のために提供されるソフトウェアモジュールである。ただし、電子原稿ライタ102は特定の出力デバイスを目的としておらず、後述の製本アプリケーション104やプリンタドライバ106により処理可能な形式に出力コマンドを変換する。この電子原稿ライタ102による変換後の形式（以後、「電子原稿形式」と呼ぶ。）は、ページ単位の原稿を詳細な書式をもって表現可能であれ

ば特に問わない。実質的な標準形式のうちでは、例えばアドビシステムズによるPDF形式や、SVG形式などが電子原稿形式として採用できる。

【0023】アプリケーション101により電子原稿ライタ102を利用する場合には、出力に使用するデバイスドライバとして電子原稿ライタ102を指定してから印刷を実行させる。ただし、電子原稿ライタ102によって作成されたままの電子原稿ファイルは、電子原稿ファイルとして完全な形式を備えていない。そのため、デバイスドライバとして電子原稿ライタ102を指定するのは製本アプリケーション104であり、その管理下でアプリケーションデータの電子原稿ファイルへの変換が実行される。製本アプリケーション104は、電子原稿ライタ102が生成した新規の不完全な電子原稿ファイルを後述する形式を備えた電子原稿ファイルとして完成させる。以下では、この点を明瞭に識別する必要がある際には、電子原稿ライタ102によって作成されたファイルを電子原稿ファイルと呼び、製本アプリケーションによって構造を与えられた電子原稿ファイルをブックファイルと呼ぶ。また、特に区別する必要がない場合は、アプリケーションにより生成されるドキュメントファイル、電子原稿ファイル、及びブックファイルをいずれも文書ファイル（または文書データ）と呼ぶ。

【0024】このようにデバイスドライバとして電子原稿ライタ102を指定し、一般アプリケーション101によりそのデータを印刷させることで、アプリケーションデータはアプリケーション101によって定義されたページ（以後、「論理ページ」あるいは「原稿ページ」と呼ぶ。）を単位とする電子原稿形式に変換され、電子原稿ファイル103としてハードディスクなどの記憶媒体に格納される。なお、ハードディスクは、本実施形態の文書処理システムを実現するコンピュータが備えているローカルドライブであってもよいし、ネットワークに接続されている場合にはネットワーク上に提供されるドライブであっても良い。

【0025】製本アプリケーション104は、電子原稿ファイルあるいはブックファイル103を読み込み、それを編集するための機能を利用者に提供する。ただし製本アプリケーション104は、各ページの内容を編集する機能は提供しておらず、ページを最小単位として構成される、後述する章やブックの構造を編集するための機能を提供している。

【0026】製本アプリケーション104によって編集されたブックファイル103を印刷する際には、製本アプリケーション104によって電子原稿デスプーラ105が起動される。電子原稿デスプーラ105は、製本アプリケーションと共にコンピュータ内にインストールされるプログラムモジュールであり、製本アプリケーションで利用するドキュメント（ブックファイル）を印刷する際に、プリンタドライバへ描画データを出力するため

に使用されるモジュールである。電子原稿デスプーラ105は、指定されたブックファイルをハードディスクから読み出し、ブックファイルに記述された形式で各ページを印刷するために、前述したOSの出力モジュールに適合する出力コマンドを生成し、不図示の出力モジュールに出力する。その際に、出力デバイスとして使用されるプリンタ107のドライバ106がデバイスドライバとして指定される。出力モジュールは、指定されたプリンタ107のプリンタドライバ106を用いて受信した出力コマンドを、プリンタ107で解釈実行可能なデバイスコマンドに変換する。そしてデバイスコマンドはプリンタ107に送信され、プリンタ107によってコマンドに応じた画像が印刷される。

【0027】<本実施形態の文書処理システムのハードウェア構成例>図2は、本実施形態の文書処理システムのハードウェア構成を示す図である。図2において、ホストコンピュータ100は、ROM203内のプログラム用ROMあるいは外部メモリ211に記憶された文書処理プログラム等に基づいて図形、イメージ、文字、表（表計算等を含む）等が混在した文書処理を実行するCPU201を備え、システムバス204に接続される各デバイスをCPU201が総括的に制御する。また、このROM203内のプログラム用ROMあるいは外部メモリ211には、CPU201の制御プログラムであるオペレーティングシステムプログラム等を記憶し、ROM203内のフォント用ROMあるいは外部メモリ211には上記文書処理の際に使用するフォントデータ等を記憶し、ROM203内のデータ用ROMあるいは外部メモリ211には上記文書処理等を行う際に使用する各種データを記憶する。RAM202は、CPU201の主メモリ、ワークエリア等として機能する。

【0028】キーボードコントローラ（KBC）205は、キーボード209や不図示のポインティングデバイスからのキー入力を制御する。CRTコントローラ（CRT）206は、CRTディスプレイ（CRT）210の表示を制御する。207はディスクコントローラ（DKC）で、ブートプログラム、各種のアプリケーション、フォントデータ、ユーザファイル、叢集ファイル、プリンタ制御コマンド生成プログラム（以下プリンタドライバ）等を記憶するハードディスク（HD）、フロッピー（登録商標）ディスク（FD）等の外部メモリ211とのアクセスを制御する。プリンタコントローラ（PRTC）208は、双方向性インタフェース（インターフェース）21を介してプリンタ107に接続されて、プリンタ107との通信制御処理を実行する。NC212はネットワークに接続されて、ネットワークに接続された他の機器との通信制御処理を実行する。

【0029】なお、CPU201は、例えばRAM202上に設定された表示情報RAMへのアウトライントの展開（ラスタライズ）処理を実行し、CRT21

0上のWYSIWYGを可能としている。また、CPU201は、CRT210上の不図示のマウスカーソル等で指示されたコマンドに基づいて登録された種々のウインドウを開き、種々のデータ処理を実行する。ユーザは印刷を実行する際、印刷の設定に関するウインドウを開き、プリンタの設定や、印刷モードの選択を含むプリンタドライバに対する印刷処理方法の設定を行える。プリンタ107は、CPU312により制御される。プリンタのCPU312は、ROM313内のプログラム用ROMに記憶された制御プログラム等あるいは外部メモリ314に記憶された制御プログラム等に基づいてシステムバス315に接続される印刷部（プリンタエンジン）317に出力情報としての画像信号を出力する。また、このROM313内のプログラムROMには、CPU312の制御プログラム等を記憶する。ROM313内のフォント用ROMには上記出力情報を生成する際に使用するフォントデータ等が記憶され、ROM313内のデータ用ROMには、ハードディスク等の外部メモリ314がないプリンタの場合には、ホストコンピュータ上で利用される情報等が記憶されている。CPU312は入力部318を介してホストコンピュータとの通信処理が可能となっており、プリンタ内の情報等をホストコンピュータ100に通知できる。RAM319は、CPU312の主メモリや、ワークエリア等として機能するRAMで、図示しない増設ポートに接続されるオプションRAMによりメモリ容量を拡張することができるよう構成されている。なお、RAM319は、出力情報展開領域、環境データ格納領域、NVRAM等に用いられる。前述したハードディスク（HD）、ICカード等の外部メモリ314は、メモリコントローラ（MC）20によりアクセスを制御される。外部メモリ314は、オプションとして接続され、フォントデータ、エミュレーションプログラム、フォームデータ等を記憶する。また、321は前述した操作パネルで、操作のためのスイッチ及びLED表示器等が配されている。また、前述した外部メモリ314は1個に限らず、複数個備えられ、内蔵フォントに加えてオプションカード、言語系の異なるプリンタ制御言語を解釈するプログラムを格納した外部メモリを複数接続できるように構成されていてもよい。更に、図示しないNVRAMを有し、操作パネル321からのプリンタモード設定情報を記憶するようにしてもよい。

<電子原稿データの形式例>編集アプリケーション104の詳細を旨とする前に、ブックファイルのデータ形式を説明する。ブックファイルは紙媒体の書物を模した3層の層構造を有する。上位層は「ブック」と呼ばれ、1冊の本を模しており、その本全般に係る属性が定義されている。その下の中間層は、本でいう章に相当し、やはり「章」と呼ばれる。各章についても、章ごとの属性が定義できる。下位層は「ページ」であり、アプリケーシ

ヨンプログラムで定義された各ページに相当する。各ページについてもページごとの属性が定義できる。ひとつのブックは複数の章を含んでいてよく、また、ひとつの章は複数のページを含むことができる。

【0030】図3は、ブックファイルの形式の一例を模式的に示す図である。この例のブックファイルにおける、ブック、章、ページは、それぞれに相当するノードにより示されている。ひとつのブックファイルはひとつのブックを含む。ブック、章は、ブックとしての構造を定義するための概念であるから、定義された属性値と下位層へのリンクとをその実体として含む。ページは、アプリケーションプログラムによって出力されたページごとのデータを実体として有する。そのため、ページは、その属性値のほか、原稿ページの実体（原稿ページデータ）と各原稿ページデータへのリンクを含む。尚、紙媒体等に出力する際の印刷ページは複数の原稿ページを含む場合がある。この構造に関してはリンクによって表示されず、ブック、章、ページ各階層における属性として表示される。

【0031】図3では、ブックファイルが1つの完結したブックである必要はないので、「ブック」を「文書」として一般化して記載している。

【0032】まず最上位に文書情報401を持つ。文書情報401は402～404の3つのパートに大別できる。文書制御情報402は、文書ファイルのファイルシステムにおけるパス名などの情報を保持する。文書設定情報403は、ページレイアウトなどのレイアウト情報とステイプルなど印刷装置の機能設定情報を保持し、ブックの属性に相当する。章情報リスト404は、文書を構成している章の集合をリスト形式で保持する。リストが保持するのは章情報405である。

【0033】章情報405も406～408の3つのパートに大別できる。章制御情報406は、章の名称などの情報を保持する。章設定情報407は、その章特有のページレイアウトやステイプルの情報を保持し、章の属性に相当する。章ごとに設定情報をもつことで最初の章は2UPのレイアウトその他の章は4UPのレイアウトのように複雑なレイアウトを持った文書を作成することが可能である。ページ情報リスト408は各章を構成する原稿ページの集合リスト形式で保持している。ページ情報リスト408が指示するのは、ページ情報データ409である。

【0034】ページ情報データ409も410～412の3つのパートに大別される。ページ制御情報410は、ツリー上に表示するページ番号などの情報を保持する。ページ設定情報411は、ページ回転角やページの配置位置情報などの情報を保持し、原稿ページの属性に相当する。ページリンク情報412は、ページに対応する原稿データである。この例では、ページ情報409が直接原稿データを持つのではなく、リンク情報412だ

けをもち、実際の原稿データは、ページデータリスト413で保持する構成としている。

【0035】図4は、ブック属性（文書設定情報403）の例を示すリストである。通常、下位層と重複して定義可能な項目に関しては、下位層の属性値が優先採用される。そのため、ブック属性にのみ含まれる項目に関しては、ブック属性に定義された値はブック全体を通して有効な値となる。しかし、下位層と重複する項目については、下位層において定義されていない場合における既定値としての意味を有する。しかし、本例では、後述するように、下位層の属性値を優先するか否かが選択可能となっている。なお、図示された各項目は具体的に1項目に対応するのではなく、関連する複数の項目を含むものもある。

【0036】ブック属性に固有の項目は、印刷方法、製本詳細、表紙／裏表紙、インデックス紙、合紙、章区切りの6項目である。これらは、ブックを通して定義される項目である。印刷方法属性としては、片面印刷、両面印刷、製本印刷の3つの値を指定できる。製本印刷とは、別途指定する枚数の用紙を束にして2つ折りにし、その束をつづり合わせることで製本が可能となる形式で印刷する方法である。製本詳細属性としては、製本印刷が指定されている場合に、見開き方向や、束になる枚数等が指定できる。

【0037】表紙／裏表紙属性は、ブックとしてまとめられる電子原稿ファイルを印刷する際に、表紙及び裏表紙となる用紙を付加することの指定、及び付加した用紙への印刷内容の指定を含む。インデックス紙属性は、章の区切りとして、印刷装置に別途用意される耳付きのインデックス紙の挿入の指定及びインデックス（耳）部分への印刷内容の指定を含む。この属性は、印刷用紙とは別に用意された用紙を所望の位置に挿入するインサート機能を持ったインサークルが使用する印刷装置に備えられている場合か、あるいは、複数の給紙カセットを使用可能である場合に有効となる。これは合紙属性についても同様である。

【0038】合紙属性は、章の区切りとして、インサークルからあるいは給紙カセットから供給される用紙の挿入の指定、及び、合紙を挿入する場合には、給紙元の指定などを含む。

【0039】章区切り属性は、章の区切り目において、新たな用紙を使用するか、新たな印刷ページを使用するか、特に何もしないか等の指定を含む。片面印刷時には新たな用紙の使用と新たな印刷ページの使用とは同じ意味を持つ。両面印刷時には、「新たな用紙の使用」を指定すれば連続する章が1枚の用紙に印刷されることはないが、「新たな印刷ページの使用」を指定すれば、連続する章が1枚の用紙の表裏に印刷されることがあり得る。

【0040】図5は、章属性（章設定情報407）の、

図6はページ属性（ページ設定情報411）の例を示すリストである。章属性とページ属性との関係もブック属性と下位層の属性との関係と同様である。

【0041】章属性に関しては、章に固有の項目はなく、すべてブック属性と重複する。したがって、通常は、章属性における定義とブック属性における定義とが異なれば、章属性で定義された値が優先する。しかし、本例では、後述するように、下位層の属性値を優先するか否かが選択可能となっている。

【0042】ブック属性と章属性とにのみ共通する項目は、用紙サイズ、用紙方向、N-up印刷指定、拡大縮小、排紙方法の5項目である。このうち、N-up印刷指定属性は、1印刷ページに含まれる原稿ページ数を指定するための項目である。指定可能な配置としては、1×1や1×2、2×2、3×3、4×4などがある。排紙方法属性は、排出した用紙にステイブル処理を施すか否かを指定するための項目であり、この属性の有効性は使用する印刷装置がステイブル機能を有するか否かに依存する。

【0043】ページ属性に固有の項目には、ページ回転属性、ズーム、配置指定、アノテーション、ページ分割などがある。ページ回転属性は、原稿ページを印刷ページに配置する際の回転角度を指定するための項目である。ズーム属性は、原稿ページの変倍率を指定するための項目である。変倍率は、仮想論理ページ領域のサイズを100%として指定される。仮想論理ページ領域とは、原稿ページを、N-up等の指定に応じて配置した場合に、1原稿ページが占める領域である。例えば1×1であれば、仮想論理ページ領域は1印刷ページに相当する領域となり、1×2であれば、1印刷ページの各辺を約70パーセントに縮小した領域となる。

【0044】ブック、章、ページについて共通な属性として、ウォーターマーク属性及びヘッダ・フッタ属性がある。ウォーターマークとは、アプリケーションで作成されたデータに重ねて印刷される、別途指定される画像や文字列などである。ヘッダ・フッタは、それぞれ各ページの上余白及び下余白に印刷されるウォーターマークである。ただし、ヘッダ・フッタには、ページ番号や日時など、変数により指定可能な項目が用意されている。なお、ウォーターマーク属性及びヘッダ・フッタ属性において指定可能な内容は、章とページとは共通であるが、ブックはそれらと異なっている。ブックにおいてはウォーターマークやヘッダ・フッタの内容を設定できるし、また、ブック全体を通してどのようにウォーターマークやヘッダ・フッタを印刷するかを指定することができる。一方、章やページでは、その章やページにおいて、ブックで設定されたウォーターマークやヘッダ・フッタを印刷するか否かを指定できる。

【0045】<本実施形態の文書処理システムの操作手順例>

（ブックファイルの生成手順例）ブックファイルは、上述したような構造及び内容を有している。次に、製本アプリケーション104及び電子原稿ライタ102によってブックファイルを作成する手順を説明する。ブックファイルの作成は、製本アプリケーション104によるブックファイルの編集操作の一環として実現される。

【0046】図7は、製本アプリケーション104によりブックファイルを開く際の手順を示すフローチャートである。

【0047】まず、開こうとするブックファイルが、新規作成すべきものであるか、それとも既存のものであるか判定する（ステップS701）。新規作成の場合は、章を含まないブックファイルを新規に作成する（ステップS702）。新規に作成されるブックファイルは、図3の例で示せば、ブックノード301のみ有し、章のノードに対するリンクが存在しないブックのノードとなる。ブック属性は、新規作成用としてあらかじめ用意された属性のセットが適用される。そして、新規ブックファイルを編集するためのユーザインタフェース（UI）画面を表示する（ステップS704）。図8は、新規にブックファイルが作成された際のUI画面の一例である。この場合には、ブックファイルは実質的な内容を持たないために、UI画面800には何も表示されない。

【0048】一方、既存のブックファイルがあれば、指定されたブックファイルを開き（ステップS703）、そのブックファイルの構造、属性、内容に従ってユーザインタフェース（UI）画面を表示する。図9は、このUI画面の一例である。UI画面900は、ブックの構造を示すツリーノード901と、印刷された状態を表示するプレビューノード902とを含む。ツリーノード901には、ブックに含まれる章、各章に含まれるページが、図3のような木構造が分かるように表示される。ツリーノード901に表示されるページは原稿ページである。プレビューノード902には、印刷ページの内容が縮小されて表示される。その表示順序は、ブックの構造を反映したものとなっている。

【0049】さて、開かれたブックファイルには、電子原稿ライタ102によって電子原稿ファイルに変換されたアプリケーションデータを、新たな章として追加することができる。この機能を電子原稿インポート機能と呼ぶ。図7の手順によって新規に作成されたブックファイルに電子原稿インポートすることで、そのブックファイルには実体が与えられる。この機能は、図8あるいは図9の画面にアプリケーションデータをドラッグアンドドロップ操作することで起動される。

【0050】図10に電子原稿インポートの手順例を示すフローチャートである。

【0051】まず、指定されたアプリケーションデータを生成したアプリケーションプログラムを起動し、デバ

イストライバとして電子原稿ライタ102を指定してアプリケーションデータを印刷出力させることで、電子原稿データに変換する（ステップS801）。変換を終えたなら、変換されたデータが画像データであるか否かを判定する（ステップS802）。この判定は、ウインドウズOSの下であれば、アプリケーションデータのファイル拡張子に基づいて行うことができる。例えば、拡張子が「.bmp」であればウインドウズビットマップデータであり、「.jpg」であればJPEG圧縮された画像データ、「.tif」であればtif形式の画像データであると判定できる。また、このような画像データの場合はS801のようにアプリケーションを起動せずに、画像データから直接電子原稿ファイルを生成することが可能であるため、S801の処理を省略することも可能である。

【0052】画像データでなかった場合には、ステップS801で生成された電子原稿ファイルを、現在開かれているブックファイルのブックに、新たな章として追加する（ステップS803）。通常、章属性としては、ブック属性と共通するものについてはブック属性の値がコピーされ、そうでないものについては、あらかじめ用意された規定値に設定されるが、本例では、後述するように、下位層の属性値を優先するか否かが選択可能となっている。

【0053】画像データである場合には、原則として新たな章は追加されず、指定されている章に、ステップS801で生成された電子原稿ファイルに含まれる各原稿ページが追加される（ステップS804）。ただし、ブックファイルが新規作成されたファイルであれば、新たな章が作成されて、その章に属するページとして電子原稿ファイルの各ページが追加される。通常、ページ属性は、上位層の属性と共通のものについてはその属性値が与えられ、アプリケーションデータにおいて定義された属性を電子原稿ファイルに引き継いでいるものについてはその値が与えられる。例えば、N-up指定などがアプリケーションデータにおいてされていた場合には、その属性値が引き継がれる。このようにして、新規なブックファイルが作成され、あるいは、新規な章が追加される。しかし、本例では、後述するように、下位層の属性値を優先するか否かが選択可能となっている。

【0054】図11は、図10のステップS801において、電子原稿ライタ102により電子原稿ファイルを生成させる手順を示すフローチャートである。

【0055】まず、新たな電子原稿ファイルを作成してそれを聞く（ステップS901）。指定したアプリケーションデータに対応するアプリケーションを起動し、電子原稿ライタ102をデバイスドライバとして、OSの出力モジュールに対して出力コマンドを送信させる。出力モジュールは、受信した出力コマンドを電子原稿ライタによって電子原稿形式のデータに変換し、出力する

（ステップS902）。出力先はステップS901で開いた電子原稿ファイルである。指定されたデータすべてについて変換が終了したか判定し（ステップS903）、終了していれば電子原稿ファイルを閉じる（ステップS904）。電子原稿ライタ102によって生成される電子原稿ファイルは、図3に示した、原稿ページデータの実体を含むファイルである。

【0056】（ブックファイルの編集例）以上のようにして、アプリケーションデータからブックファイルを作成することができる。生成されたブックファイルについては、章及びページに対して次のような編集操作が可能である。

- (1) 新規追加
- (2) 削除
- (3) コピー
- (4) 切り取り
- (5) 貼り付け
- (6) 移動
- (7) 章名称変更
- (8) ページ番号名称振り直し
- (9) 表紙挿入
- (10) 合紙挿入
- (11) インデックス紙挿入
- (12) 各原稿ページに対するページレイアウト。

【0057】このほか、一旦行った編集操作を取り消す操作や、さらに取り消した操作をやり直す操作が可能である。これら編集機能により、例えば複数のブックファイルの統合、ブックファイル内で章やページの再配置、ブックファイル内で章やページの削除、原稿ページのレイアウト変更、合紙やインデックス紙の挿入などといった編集操作が可能となる。これらの操作を行うと、図4乃至図6に示す属性に操作結果が反映されたり、あるいはブックファイルの構造に反映される。たとえば、ブランクページの新規追加操作を行えば、指定された箇所にブランクページが挿入される。このブランクページは原稿ページとして扱われる。また、原稿ページに対するレイアウトを変更すれば、その変更内容は、印刷方法やN-up印刷、表紙／裏表紙、インデックス紙、合紙、章区切りといった属性に反映される。

【0058】本例における、編集時の表示及び操作例については、以下で詳細に示す。

【0059】（ブックファイルの出力例）以上のように作成・編集されるブックファイルは印刷出力を最終目的としている。利用者が図9に示す製本アプリケーションのUI画面900からファイルメニューを選択し、そこから印刷を選択すると、指定した出力デバイスにより印刷出力される。この際、まず製本アプリケーション104は、現在開かれているブックファイルからジョブチケットを作成して電子原稿デスプーラ105に渡す、電子原稿デスプーラ105は、ジョブチケットをOSの出力

コマンド、例えばウインドウズのGDIコマンドに変換し、それを出力モジュール、例えばGDIに送信する。出力モジュールは、指定されたプリントドライバ106によってデバイスに適したコマンドを生成し、そのデバイスに送信する。

【0060】すなわち、図示しない出力モジュールのグラフィックエンジンは、印刷装置ごとに用意されたプリントドライバ106を外部メモリ211からRAM202にロードし、出力をプリントドライバ106に設定する。そして、出力モジュールは、受け取ったGDI(Graphic Device Interface)関数からDDI(Device Driver Interface)関数に変換して、プリントドライバ106へDDI関数を出力する。プリントドライバ106は、出力モジュールから受け取ったDDI関数に基づいて、プリンタが認識可能な制御コマンド、例えばPDL(Page Description Language)に変換する。変換されたプリンタ制御コマンドは、OSによってRAM202にロードされたシステムスプーラを経てインタフェース211経由でプリント107へ印刷データとして出力される仕組みとなっている。

【0061】ここで、上記ジョブチケットは、原稿ページを最小単位とする構造を有するデータである。ジョブチケットにおける構造は、用紙上における原稿ページのレイアウトを定義している。ジョブチケットは1ジョブにつき1つ発行される。そのため、まず最上位にドキュメントというノードがあり、文書全体の属性、例えば両面印刷／片面印刷などが定義されている。その下には、用紙ノードが属し、用いるべき用紙の識別子や、プリントにおける給紙口の指定などの属性が含まれる。各用紙ノードには、その用紙で印刷されるシートのノードが属する。1シートは1枚の用紙に相当する。各シートには、印刷ページ(物理ページ)が属する。片面印刷ならば1シートには1物理ページが属し、両面印刷ならば1シートに2物理ページが属する。各物理ページには、その上に配置される原稿ページが属する。また物理ページの属性として、原稿ページのレイアウトが含まれる。

【0062】ジョブチケットのデータ構造の例を図12に示す。印刷用のデータでは、文書は用紙の集合で構成されており、各用紙は表、裏の2面で構成されており、各面は原稿をレイアウトする領域(物理ページ)を持ち、各物理ページには、最小単位である原稿ページの集合から構成される。1101は文書に相当するデータで、文書全体に関係するデータと、文書を構成する用紙情報のリストから構成される。用紙情報1102は用紙サイズなど用紙に関する情報と用紙上に配置される面情報のリストから構成される。面情報1103は、面に固有のデータと、面上に配置される物理ページのリストから構成される。物理ページ情報1104は、物理ページのサイズやヘッダ・フッタなどの情報と、物理ページを構成する原稿ページのリストから構成される。

【0063】電子原稿デスプーラ105は、上述のジョブチケットを、出力モジュールへの出力コマンドに変換する。

【0064】(プレビュー表示の内容例)すでに説明したとおり、ブックファイルが製本アプリケーションによって開かれると、図9に示すユーザインターフェース画面900が表示される。ツリー部901には、開いているブック(以下、「注目ブック」と呼ぶ。)の構造を示すツリーが表示される。プレビュー部には、利用者の指定に応じて、3通りの表示方法が用意されている。第1は原稿ページをそのまま表示する原稿ビューと呼ばれるモードである。原稿ビューモードでは、注目ブックに属する原稿ページの内容が縮小されて表示される。プレビュー部の表示にレイアウトは反映されない。第2は印刷ビューモードである。印刷ビューモードでは、プレビュー部902には、原稿ページのレイアウトが反映された形で原稿ページが表示される。第3は簡易印刷ビューモードである。簡易印刷ビューモードでは、各原稿ページの内容はプレビュー部の表示には反映されず、レイアウトのみが反映される。

【0065】<他の文書処理システムの構成例>上記実施形態の文書処理システムはスタンドアロン型のシステムであるが、これを拡張したサーバクライアントシステムでもほぼ同様の構成・手順でブックファイルが作成・編集される。ただし、ブックファイルや印刷処理はサーバによって管理される。

【0066】図13はサーバクライアント型文書処理システムの構成を示すブロック図である。

【0067】クライアント文書処理システムは、スタンドアロン型システムに、クライアントモジュールであるDOMS(Document Output Management Service:文書出力管理サービス)ドライバ109及びDOMSプリントサービスモジュール110、DS(文書サービス)クライアントモジュール108を加えた構成を有する。このクライアント文書処理システム1200に、文書管理サーバ1201及び印刷集中管理サーバ1202及びプリントサーバ1203が接続されている。これらサーバは、通常ネットワークによってクライアント文書処理システムと接続されるが、サーバが同時にクライアントとしても機能する場合には、ネットワーク間の通信をシミュレートするプロセス間通信によって接続される。尚、図13では文書管理サーバ1201と印刷集中管理サーバ1202の両サーバがクライアントに接続されているが、いずれか一方のみがネットワーク上に存在する場合もあり得る。接続されているサーバが文書管理サーバであれば、そのクライアントモジュールを含む文書管理サーバクライアントシステム1201SCが、印刷集中管理サーバ1202であれば、そのクライアントモジュールを含む印刷管理サーバクライアントシステム1202SCが、スタンドアロン型文書管理システムに追加され

る。

【0068】文書管理サーバ1201は、製本アプリケーション104により作成・編集されたブックファイルを格納するサーバである。文書管理サーバ1201によってブックファイルを管理する場合、ブックファイルは、クライアントPCのローカルHDに代わって、あるいはそれに加えて、文書管理サーバ1201のデータベース1211に保存される。製本アプリケーション104と文書管理サーバ1201との間のブックファイルの保存及び読み出しが、DSクライアント108及びDSコア1212を介して行われる。

【0069】印刷集中管理サーバ1202は、クライアント文書管理システム1200に格納された、あるいは文書管理サーバ1201に格納されたブックファイルの印刷を管理するサーバである。クライアントにおける印刷要求は、DOMSドライバ109及びDOMSプリントサービスモジュール110を介して印刷集中管理サーバ1202のDOMSWGサーバモジュール1221に送信される。集中印刷管理サーバ1202は、クライアントのプリンタで印刷する場合にはクライアントのDOMSプリントサービスモジュール110を介して電子原稿デスプーラ105に電子原稿データを渡し、プリントサーバ1203により印刷する場合には、プリントサーバ1203のDOMSプリントサービスモジュール1203に送信する。集中印刷管理サーバは、例えば保存されているブックファイルに対して印刷要求を発行した利用者の資格などについてセキュリティチェックを行ったり、印刷処理のログを保存したりする。このように、文書処理システムは、スタンダードアロンとしても、クライアントサーバシステムとしても実現できる。

【0070】[本実施形態の文書処理システムの編集操作例]上記図9は、製本アプリケーション104の操作画面である。製本アプリケーションは、文書のページ順の入れ替えや複製、削除などの編集に加え、ステイブルなどの印刷装置の機能設定も行うことができ、指定した印刷装置に印刷することができる。図9の左側の領域には、文書の構造を示すツリービューが表示されている。文書は、章の集合で構成されており、各章は原稿ページの集合で構成されている。図9の右側の領域には、各ページの印刷プレビューが表示されている。

【0071】<本実施形態の文書処理システムの属性設定例>図14は、製本アプリケーション104の「文書の詳細設定」ウインドウ1400を示している。

【0072】このウインドウでは、「文書設定情報403」を表示／設定することができる。このウインドウは、図9のアプリケーション操作画面にあるPrint Formメニューの「Detailed Setting for Document」メニューあるいはツールバー上の「Detailed Setting for Document」ボタンから起動される。「文書の詳細設定」ウインドウは、文書全体に影響する属性の設定を行うため

のウインドウである。このウインドウは、Page Setup、Decoration、Edit、Paper Sourceの4つのシートから構成されており、図14は、Page Setupシートを表示した状態を示している。このPage Setupシートでは、主にレイアウトに関する設定を行うことができ、用紙サイズや向き、Nページ印刷などの設定を指示することができる。このウインドウには、ズームに関するチェックボックスコントロール1401、1402が配置されている。

【0073】図15は、製本アプリケーション104の「章の詳細設定」ウインドウ1500を示している。

【0074】このウインドウでは、「章設定情報407」を表示／設定することができる。このウインドウは、図9のアプリケーション操作画面にあるPrint Formメニューの「Detailed Setting for Chapter」メニューあるいはツールバー上の「Detailed Setting for Chapter」ボタンから起動される。「章の詳細設定」ウインドウは、章固有の属性の設定を行うためのウインドウである。このウインドウは、Page Setup、Decoration、Edit、Paper Sourceの4つのシートから構成されており、図15は、Page Setupシートを表示した状態を示している。このPage Setupシートでは、主に各章で固有のレイアウトに関する設定を行うことができ、用紙サイズや向き、Nページ印刷などの設定を指示することができる。

「文書の詳細設定」と「章の詳細設定」で重複する設定項目については「Follow Book Attribute」のチェックボックスコントロール1501、1502、1503を配置している。このチェックボックスにチェックが入っている項目グループについては、文書の設定値をその章にも適用する。このチェックをはずした場合については、次の図16に示す。章固有の設定としては2種類に分類できる。一つは章でしか持たない設定項目である。もう一つは、上位階層の文書とは異なる独自の設定値を章の階層で保持している場合である。

【0075】図16は、図15の「章の詳細設定」でチェックボックスコントロール1501、1502のチェックをはずした状態である。この場合、文書の全体がA4用紙を使っている場合でも、この章を構成するページはA4用紙サイズとする。またレイアウトも、文書では1 Page Per Sheetであっても、この章は4 Page Per Sheetでレイアウトするという指定である。「MS Arrange」については、チェックがついているので、上位階層である文書の設定値をこの章の設定値として使う。

【0076】図17は、製本アプリケーション104の「ページの詳細設定」ウインドウ1700を示している。

【0077】このウインドウでは、「ページ設定情報411」を表示／設定することができる。このウインドウは、図9のアプリケーション操作画面にあるPrint Formメニューの「Detailed Setting for Page」メニューあ

るいはツールバー上の「Detailed Setting for Page」ボタンから起動される。「ページの詳細設定」ウインドウは、各ページで固有の属性の設定を行うためのウインドウである。このウインドウは、Page Setup、Editの2つのシートから構成されており、図17は、PageSetupシートを表示した状態を示している。このPage Setupシートでは、主に各ページで固有のレイアウトに関する設定を行うことができ、原稿ページを配置する際の回転角や拡縮率などの設定を指示することができる。「章の詳細設定」と「ページの詳細設定」で重複する設定項目については「Follow Chapter Attribute」のチェックボックスコントロール1701を配置している。このチェックボックスにチェックが入っている項目については、章の設定値をそのページにも適用する。このチェックをはずした場合については、次の図18に示す。

【0078】図18は、図17の「ページの詳細設定」でPage Rotationの設定を変更した状態を示している。コントロール1701で、原稿ページをレイアウトする際にページの上が左になるように回転して配置する設定にしている。この項目については「Follow Chapter Attribute」のチェックボックスが無いので、章や文書に重複する設定項目が無いので、このウインドウに表示されている設定が常にページの設定値となる。1701のチェックがある場合、例えば「Arrange」の項目は、章にも重複する設定項目があるので章の設定値をページの設定値とする。章で該当する「Arrange」の項目が「Follow Book Attribute」のチェックがある場合、章の設定値は上位階層である文書の設置値を使うので結局ページの設定値は文書の設定値を使うことになる。章で該当する項目のチェックが外れている場合、章固有の設定値を持っているので、ページの設定値はその章固有の設定値を使う。

【0079】尚、上記設定画面で設定されたチェックボックスコントロールの情報は、専用の領域に保持されてもよいが、図3に示した設定情報内に属性の1つとして保持するのが望ましい。この場合には、図5及び図6にチェックボックスコントロールの情報を保持する領域が追加される。

【0080】<本実施形態の文書処理システムの文書編集の表示例>上位階層と重複する設定項目を上位階層の設定値を使わないようにした場合、つまり、チェックボックスコントロール1501や1701のチェックをはずした場合のアプリケーションの表示形式を、図19に示す。

【0081】図19の例では、文書は二つの章からなり、各章は9ページの原稿データを持っている。最初の章のレイアウトを4UP(4in1とも表記する)に、2番目の章の先頭ページ(文書全体では10ページ目)に回転を指定した場合の表示例である。右側のプレビューにおいて、1~3ページ目には1ページ上に4つの原

稿がそれぞれ配置されている様子が表示されている。そして、第4ページ目のデータである「A」の文字が回転している様子が表示されている。ツリービューでの章やページ固有の設定を行った場合の表示形式を説明する。最初の章に対するアイコン1901が変化して、この章に章独自の設定がなされていることを示している。2番目の章の先頭ページに対するアイコン1902が変化して、このページに特別な設定がなされていることが示されている。

【0082】図20及び図21は、本システムにおいてファイル全体の挿入、一部のコピーペーストを行う場合の動作を説明する図である。

【0083】図20Aは文書A、図20Bは文書Bを示している。文書Aに対して、文書Bから章C2001をコピーペーストする場合を考える。文書Aは2ページ印刷、文書Bは4ページ印刷とする。

【0084】図20A乃至図20Dは、章Cに章固有の設定がなされていない場合の例である。章Cをペーストした結果には2通りの考え方方が可能である。1. 章Cは所属する文書の設定に従う。この場合、章C2002に示される通りペースト先である文書Aの設定である2ページ印刷の設定でレイアウトされる(図20C)。2. 章Cはもともとあった文書Bの設定である4ページのレイアウトを保存する。この場合、章Cは2003に示される通りに章固有の設定として4ページ印刷の設定値を保持し、文書の設定とは無関係に4ページ印刷でレイアウトされる(図20D)。

【0085】図21A乃至図21Eは、章Cに章固有の設定がなされてる場合の例である。図21Bにあるように、章C2101が1ページ印刷の設定がなされているとする。この場合、文書Aにペーストした場合の結果として3通り考えられる。

1. 章C2102は、やはり文書Aの設定を受け継ぎ2ページ印刷の設定となる(図21C)。

2. 章C2103は、章固有の設定値を保持し、文書A中においても1ページ印刷の設定を保持しつづける(図21D)。

3. 章C2104は、もともとの文書Bの設定を受け継ぎ4ページ印刷の設定を保持する。この場合、文書Aは2ページ印刷だが章Cの部分は1ページ印刷となる(図21E)。

【0086】ただし、3番目は見た目に出でこない文書Bの設定を保持するのはわかりにくいので通常使われることは考えにくい。

【0087】<本実施形態の文書処理システムの文書編集の動作手順例>次に、ファイル全体の挿入、一部分のコピーペーストを行う場合の処理フローについて説明する。

【0088】(コピー手順) 図22は、コピーペーストのコピー時の処理概要を説明したフローチャートであ

る。コピーペーストする場合、コピー時にどの形式でコピーするかを決めてしまう方式と、コピー時はできるだけ多くの情報をコピーし、ペースト時にいろいろな形式を選択できるようにする方法がある。この例は後者の方で、ペースト時のペースト形式の選択肢を増やすために、ペースト処理に不要のデータ以外はできるだけコピーしている。図3の文書データ構造の階層構造を保ってほぼそのままの形でコピーすることでペースト時の選択を可能としている。

【0089】まず、ステップS1401で、挿入元となる文書の文書情報401をコピーする。ファイル名など不要な部分はコピーしないほうが効率がよい。次に、ステップS1402へ進み必要な章を404の章リストへ追加する。文書ファイルの追加の場合は、全ての章が追加対象である。コピーペーストやマウスの操作によるDrag&Dropの場合は、選択されている部分の章が追加対象となる。章が選択されている場合は選択されている章全て、章ではなくページが選択されている場合は選択されているページを含む章が追加対象である。次に、ステップS1403へ進み、ステップS1402で追加した章の章情報を挿入元の文書の対応する章からコピーする。この場合も不要な情報をコピーしないことで効率化を図ることが可能である。次に、ステップS1404へ進み、各章を構成するページを1402でコピーした章情報のページリスト部へ追加する。ここでも追加するページは必要なページだけでよい。次に、ステップS1405へ進み、各ページに対応するページ情報を挿入元文書からコピーする。

【0090】(ペースト手順)図23は、文書全体をペースト(文書の挿入)する場合の処理のフローチャートである。

【0091】まず、ステップS1501で最上位である文書の設定を下位の構成要素である章の設定へと移すかどうかを判定する。図21Eの形式でペーストする場合、元文書の設定を章の設定へコピーすることになる。ここで章の設定へ移さないと判定された場合は、上位階層の設定を下位階層に移す必要が無いので、ステップS1510へ進みコピーされているデータをそのままペーストする。ステップS1501で章設定に移す場合、ステップS1502へと進む。ステップS1502はループの終了条件を判定しており、コピーされている全ての章に対して処理が終了したかどうかを判定する。全ての章に対して処理が終了している場合はステップS1510へ進み処理済のデータをペーストする。ステップS1502で全ての章に対して処理が終了していないと判定された場合、ステップS1503へ進む。

【0092】ステップS1503では、処理が残っている章を次の処理対象とする。図22では章をリストでコピーしているので、リストの先頭から順に処理対象とすればよい。次に、ステップS1504へ進み、処理対象の章について全ての設定項目について処理が終了したか判定する。全ての項目の処理が終わった場合、ステップS1602へ戻り処理対象を次の章とする。ステップS1604で未処理の項目が残っていると判定された場合、ステップS1605へ進み、未処理の設定項目を処理対象とする。さらに、ステップS160

の章について全ての設定項目について処理が終了したか判定する。全ての項目の処理が終わった場合、ステップS1502へ戻り処理対象を次の章とする。ステップS1504で未処理の項目が残っていると判定された場合、ステップS1505へ進み、未処理の設定項目を処理対象とする。さらにステップS1506へ進み、ステップS1505で処理対象とした設定項目について、章固有の設定値を保存するか、元文書の上位階層に合わせるか、挿入先文書の設定に合わせるの3方式のいずれとするかを判断する。

【0093】ステップS1506で章固有の設定値を保存すると判定された場合、あるいは、現在の設定項目が上位階層である文書に対応する設定項目が無い場合、ステップS1507へ進み、そのまま章固有の値を保持しつづける。ステップS1506で元文書の上位の設定に合わせると判断された場合、ステップS1508へ進みコピーした上位階層の対応する設定項目の設定値で章の設定値を上書きする。ステップS1506で挿入先の設定値に合わせると判断された場合、ステップS1509へ進み、設定値を挿入先の上位階層のに合わせる設定値に変更する。ステップS1507、ステップS1508、ステップS1509のいずれかが終わると再びステップS1504へ戻り、次の設定項目を処理対象として処理を続行する。

【0094】図24は、章をペーストする場合の処理のフローチャートである。

【0095】まず、ステップS1601でコピー元と挿入先の文書が同一文書であるかを判定する。ここで同一文書内のコピーペーストであると判断された場合、上位階層の設定を下位階層に移す必要が無いので、ステップS1610へ進みコピーされているデータをそのままペーストする。ステップS1601で別文書への挿入と判断された場合、ステップS1602へと進む。ステップS1602はループの終了条件を判定しており、コピーされている全ての章に対して処理が終了したかどうかを判定する。全ての章に対して処理が終了している場合はステップS1610へ進み処理済のデータをペーストする。ステップS1602で全ての章に対して処理が終了していないと判定された場合、ステップS1603へ進む。

【0096】ステップS1603では、処理が残っている章を次の処理対象とする。図22で説明した例では章をリストでコピーしているので、リストの先頭から順に処理対象とすればよい。次に、ステップS1604へ進み、処理対象の章について全ての設定項目について処理が終了したか判定する。全ての項目の処理が終わった場合、ステップS1602へ戻り処理対象を次の章とする。ステップS1604で未処理の項目が残っていると判定された場合、ステップS1605へ進み、未処理の設定項目を処理対象とする。さらに、ステップS160

6へ進み、ステップS1605で処理対象とした設定項目について、章固有の設定値を保存するか、元文書の上位階層に合わせるか、挿入先文書の設定に合わせるの3方式のいずれとするかを判断する。

【0097】ステップS1606で章固有の設定値を保存すると判定された場合、あるいは、現在の設定項目が上位階層である文書に対応する設定項目が無い場合、ステップS1607へ進み、そのまま章固有の値を保持しつづける。ステップS1606で元文書の上位の設定に合わせると判断された場合、ステップS1608へ進みコピーした上位階層の対応する設定項目の設定値で章の設定値を上書きする。ステップS1606で挿入先の設定値に合わせると判断された場合、ステップS1609へ進み、設定値を挿入先の上位階層のに合わせる設定値に変更する。ステップS1607、ステップS1608、ステップS1609のいずれかが終わると再びステップS1604へ戻り、次の設定項目を処理対象として処理を続行する。

【0098】図25は、ページをペーストする場合の処理のフローチャートである。コピーしたのは章であっても、章としてではなく章に含まれる個別のページを挿入する場合もこの例で示す方法で処理可能である。

【0099】まず、ステップS1701でコピー元と挿入先の文書が同一文書であるかを判定する。ここで同一文書内でのコピーペーストであると判断された場合、上位階層の設定を下位階層に移す必要が無いので、ステップS1712へ進みコピーされているデータをそのままペーストする。ステップS1701で別文書への挿入と判断された場合、ステップS1702へと進む。ステップS1702はループの終了条件を判定しており、コピーされている全てのページに対して処理が終了したかどうかを判定する。全てのページに対して処理が終了している場合はステップS1710へ進み処理済のデータをペーストする。ステップS1702で全てのページに対して処理が終了していないと判定された場合、ステップS1703へ進む。

【0100】ステップS1703では、処理が残っているページを次の処理対象とする。図22で説明した例では章とページをリストでコピーしているので、章リストの先頭から順に、章のページリストの先頭から処理対象とすればよい。次に、ステップS1704へ進むみ、処理対象のページについて全ての設定項目について処理が終了したか判定する。全ての項目の処理が終わった場合、ステップS1702へ戻り処理対象を次の章とする。ステップS1704で未処理の項目が残っていると判定された場合、ステップS1705へ進み、未処理の設定項目を処理対象とする。さらにステップS1706へ進み、ステップS1705で処理対象とした設定項目について、ページ固有の設定値を保存するかを判断する。

【0101】ステップS1706でページ固有の設定値を保存すると判定された場合、あるいは、現在の設定項目が上位階層である章や文書には対応する設定項目が無い場合、ステップS1710へ進み、そのままページ固有の値を保持しつづける。ステップS1706でページ固有の設定を保存しないと判断された場合、上位階層の設定に合わせるため、ステップS1707へ進む。

【0102】ここで、上位階層である章の対応設定項目を調べ、章に固有の設定値を持っているか、さらに上位階層に従う設定となっているかを調べる。ステップS1707で章固有の設定は無いと判断された場合ステップS1708へ進み章の設定値をページの設定値へコピーする。ステップS1707で章固有の設定値が無いと判断された場合、ステップS1709へ進み上位階層の設定として章の設定と文書の設定のどちらを優先するか判断する。ステップS1709で章設定を優先すると判断した場合、ステップS1708へ進む。ステップS1709で賞を優先しないと判断した場合、文書の対応する設定項目の設定値をページにコピーする。ステップS1708、ステップS1710、ステップS1711のいずれかが終わると再びステップS1704へ戻り、次の設定項目を処理対象として処理を続行する。

【0103】ここで示した例では、各設定項目毎にどの階層の設定値を優先して用いるか、コピー元／挿入先のどちらの設定を優先するかを判断し、設定項目ごとにきめ細かく優先設定を決めることが可能である。尚、図23及び図24の例では文書と章の2階層で判定していたが、図25のページ設定の処理も組み合わせて、文書や章のペーストでもページの設定まで処理することも可能である。その場合、ステップS1501やS1601の直前に図25のフローを入れればよい。

【0104】<本実施形態の文書処理システムの文書編集の操作指示例>次に、ユーザーの文書編集操作（特に、Drag&Drop）に対応した本実施形態の表示方法を説明する。

【0105】図26は、マウス操作によるコピーペーストといわゆるDrag&Dropの操作をする場合のマウスカーソルの形態変化の例である。

【0106】左端図は、ページの外にマウスカーソルがある場合の例である。マウスカーソル2601は、通常状態のデフォルト形態である矢印型である。このままページを選択しようとして、ページの上にマウスカーソルを移動したところが左から2つ目の図である。マウスカーソル2602は手の形に変わり、ページが移動(Drag & Drop)の対象物であることを示している。ここでマウスをクリックし、選択したところが左から3つ目の図である。ページが選択対象であることを示してページが太枠で囲まれている。ここでマウスをクリックし、Dragを開始すると右端図に示すようにマウスカーソル2604はページをつかんでいることを示すように握った形とな

り、そのまま移動可能であることをユーザに示す。

【0107】(プレビューエリアでの表示例)図27は、DragしたページのDrag&Drop位置を示す表示形式の例である。

【0108】Drop位置が確認しやすいように、Drag対象のページは半透明となっている。また、Drop位置を示すために、バー2701が表示される。この図の状態でマウスボタンを離してDropすると、(1-1)と表示されたページと(1-2)と表示されたページの間に、Dragページ(1-3)が入りこみ、(1-1)、(1-3)、(1-2)の順序に並び替えがおきる。このDrag&Drop操作の際に、コントロールキーとShiftキーなどほかの操作を併用することで、ページ移動ではなくページの複製処理を行うことも可能である。また、通常の左ボタンではなく右ボタンを使うことで、Drop時に移動・複製いずれの処理を行うかを選択させることも可能である。

【0109】(ツリーエリアでの表示例)図28は、Tree表示部でのDrag&Dropの表示形式を説明した図である。

【0110】Drag中の表示はTree部の小さな表示形式に合わせて、Drag対象のページ表示は消え、マウスカーソル2801は矢印とページアイコンとの組み合わせ表示となる。マウスを動かし、マウスカーソル2801を2803の位置に動かすと、Drop位置を示すバーは2802の位置から2804の位置へとマウスカーソルの位置に合わせて移動する。

【0111】Treeの場合は、Treeの表示自体に階層情報が入っているので表示がわかりやすいが、文書が文書一章一ページの階層構造を持つため、章と章の間にマウスカーソルを移動した場合にどちらの章にDropするのかという問題が発生する。本実施形態では、図29のように、バーが2901の位置にある場合は、ページ(1-3)の次という意味で「ドキュメント1」の章の最終ページにDropされる。さらにマウスを下に移動すると「ドキュメント2」のアイコンとページ(2-1)のページの間2902の位置にバーが移動する。この位置にバーがきた場合「ドキュメント2」の章の先頭ページとしてDropされる。

【0112】(プレビューエリアでの表示変形例)一方、プレビューエリアの場合、表示形式に階層の考え方がないので分かりにくい。章の境界に位置する場所にバーが移動した場合、先頭側の章の末尾あるいは、末尾側の章の先頭ページにDrop、あるいは、マウスカーソルの移動軌跡に依存して先頭側から移動してから先頭側の末尾へ末尾側から移動してきた場合には末尾側の章の先頭へという決め方もある。

【0113】図22は、章の境界部分だけ2回バーを表示する場合の例である。“1”と書かれたページと“2”と書かれたページが章の境界だとすると、“1”的ページに

近い位置では3001の位置にバーが表示され、“2”的ページに近い場合は3002の位置にバーが表示される。バー3001が表示されている時にDropした場合には前の章の末尾は、バー3002が表示されている時にDropした場合は後の章の先頭へとDropされる。尚、2本のバー3001及び3002が表示されて、その太さや濃さ、あるいはカラー表示に場合は色でDropされる位置を表示してもよい。

【0114】(Nページ印刷の場合のプレビューエリアでの表示例)次にNページ印刷の場合の表示方法について説明する。

【0115】図31は1ページ印刷の場合の例である。この例では、用紙を示す矩形3101が原稿のページと対応しているのでDrop位置を示すバーは用紙の矩形と矩形との間に表示する形式である。

【0116】図32は4ページ印刷の場合の例である。この場合、4ページ分のデータが描画された面で一つの物理ページとして扱われる。物理ページの間にDropさせる場合、図32の左図のように物理ページを表現する矩形と矩形の間にバーを表示する。原稿のページの間にDropできる場合には、一つの物理ページの中に複数のDrop可能位置が存在する。図32の右図は“1”と書かれた原稿ページと“2”と書かれた原稿ページとの間にDrop位置を示すバーを表示している。このように、Drag処理の対象物によって切り分けて表示している。

【0117】物理ページを選択Dragしている場合、物理ページ単位で移動する処理と考え、図32の左図の形式でDropさせる。この場合の内部処理は、物理ページ上の全ての論理ページを選択してDragしているのと同じ処理をしている。そのため、選択した物理ページとDrop位置とでNページ印刷の設定が違う場合、Drop先のレイアウト指定に従ってレイアウトされるため、Drag開始時とは異なるレイアウトとなる。論理ページを選択Dragしている場合、論理ページ単位で移動すると考え図32の右図の形式でDropさせる。ただし、内部データ的には複数論理ページのDrag&Drop処理と同じであるので、マウスの位置により図32の左右の図の表示形式を切り替えて、任意の位置へのDropを許すことも可能である。

【0118】尚、本実施形態ではバーで挿入位置を報知したが、バーに限定されずユーザの操作に有用であればどのような形状(矢印等のアイコン表示)であっても構わず、これらも本発明に含まれる。

【0119】次に、Dragしている対象である章やページが、章あるいはページ固有の設定を持っている場合の例を図33に示す。

【0120】章やページに固有の設定がある場合の表示例を図9に示した。Drag中のデータに固有の設定を持つページや章が含まれてい無い場合、3301の形式の表示である。Drag中のデータに章やページ固有の設定値をもつものが含まれている場合、3302に示す形の表示

形式となる。図9の901や902のように左上に小さなマークをつけることで固有設定の有無を表現する。Dropした後も、固有の設定を保持する場合、図9に示すように固有の設定を持った章あるいはページとなる。Drop時に固有設定のクリアをするような指定を行っている場合、Drop時に内部的に固有設定をクリアする。この場合の処理のフローチャートは、既に図24及び図25に示してある。

【0121】<本実施形態の文書処理システムの操作指示の手順例>図34に、本実施形態の文書処理システムの操作指示の手順例をフローチャートで示す。尚、カーソルの位置検出や対象画像の表示位置とカーソル位置との距離検出などは既知があるので、詳細には説明しない。又、ブック、章、ページ間の設定(属性)の優先設定などの処理についても、煩雑となるので省略している。この場合は、図24や図25の処理と共に、図33のような固有の設定があることを示す表示が付加される。

【0122】まず、ステップS2001でマウス操作からDrag中か否かを判定する。Drag中でなければ、ステップS2002でDropか否かを判定する。Dropでもなければステップ2004で本例では対象としない他の処理を行なって、リターンする。Dropと判定すると、ステップS2003に進んでカーソル位置に従って表示されているバーによって示されたDrop位置に、データの移動あるいは複製が行われる。移動／複製の選択も煩雑になるので示していない。

【0123】ステップS2001でDragと判定されると、ステップS2005で既知の方法でカーソル位置を検出する。次に、ステップS2006でカーソル位置に最も近いツリー及びプレビューエリア内の原稿ページの境界を検出する。ステップS2007では、ステップS2006で検出されたカーソルに最も近い原稿ページの境界が、ツリーエリア内にあるかプレビューエリア内にあるかを判断する。ツリーエリア内にある場合は、ステップS2008で原稿ページの境界位置にDrop位置を示すバーを表示する。表示方法も既知であるので詳説はない。

【0124】プレビューエリアの場合は、まずステップS2009で検出されたカーソルに最も近い原稿ページの境界が同時に章の境界であるかが判定される。章の境界であればステップS2010で図30に示すような複数回のバー表示をするように設定する。図34には詳細な説明はしないが、複数回のバー表示が設定された場合には、原稿ページの境界位置に複数の境界が設定されて、更にカーソルとの距離が検出され、より近い位置にバーが表示される。あるいは複数のバーが、太さ、濃さ、色などにより、識別可能(より近いバーが目立つように)に表示される。章の境界でなければ何もせずにステップS2011に進む。

【0125】ステップS2011ではNページ印刷か否かが判定され、Nページ印刷のばあいは更にステップS2012で原稿ページの境界が印刷ページの境界か否かが判定される。Nページ印刷で印刷ページの境界でない場合は、ステップS2013で図32の右図のように、印刷ページ内の原稿ページの境界部分にバーを表示する。一方、Nページ印刷でない、又はNページ印刷であっても印刷ページの境界で有る場合は、図30、図31、図32の左図のように、印刷ページ外(間)にバーを表示する。

【0126】尚、本実施形態で示したフローチャートにおいて、その前後は処理が未完成にならない限り入れ替えることが可能である。

【0127】また、本発明は、複数の機器(例えばホストコンピュータ、インターフェース機器、リーダ、プリンタなど)から構成されるシステムに適用しても、一つの機器からなる装置(複写機、プリンタ、ファクシミリ装置など)に適用してもよい。

【0128】また、本発明の目的は、前述した実施形態の機能を実現するソフトウェアのプログラムコードを記憶した記憶媒体を、システムあるいは装置のコンピュータ(またはCPUやMPU)が記憶媒体に格納されたプログラムコードを読み出し実行することによって、達成される。この場合、記憶媒体から読み出されたプログラムコード自体が前述した実施形態の機能を実現することになり、そのプログラムコードを記憶した記憶媒体は本発明を構成することになる。

【0129】プログラムコードを供給するための記憶媒体としては、例えば、フロッピーディスク、ハードディスク、光ディスク、光磁気ディスク、CD-ROM、CD-R、磁気テープ、不揮発性のメモリカード、ROMなどを用いることができる。

【0130】また、コンピュータが読み出したプログラムコードを実行することにより、前述した実施形態の機能が実現されるだけでなく、そのプログラムコードの指示に基づき、コンピュータ上で稼動しているOS(オペレーティングシステム)などが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれる。

【0131】さらに、記憶媒体から読み出されたプログラムコードが、コンピュータに挿入された機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに書き込まれた後、そのプログラムコードの指示に基づき、その機能拡張ボードや機能拡張ユニットに備わるCPUなどが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれる。

【0132】

【発明の効果】本発明によれば、データと設定値を階層的に保持するような文書処理システムで、データの挿入

・移動・コピーペーストする際に、階層的にデータと設定値を保持しているという特徴を最大限に活かして、編集操作の効率化とユーザビリティを向上させる表示を実現する文書処理方法及びそのシステムを提供できる。

【0133】すなわち、Dragしているデータに固有の設定の有無を容易に確認することが可能であるという効果がある。

【0134】また、かかる構成において、同一のデータであってもDragしているエリアの表示形式に応じて、適切な表示形式に切り替えることが可能であるという効果がある。

【0135】また、かかる構成において、Dragデータの内容に応じたDrop位置を容易に理解することが可能であるという効果がある。

【0136】また、かかる構成において、Drop時に複数のDrop処理の選択肢がある場合に、Drop位置の表示形式を見ることで、Drop前にどの選択肢が選ばれるかを知ることが可能となるという効果がある。

【図面の簡単な説明】

【図1】本実施形態のスタンドアロン型の文書処理システムのソフトウェア構成例を示すブロック図である。

【図2】本実施形態の文書処理システムを実現するハードウェア構成例を示すブロック図である。

【図3】ブックファイルの構造の一例を示す図である。

【図4】ブック属性の一覧を示す図である。

【図5】章属性の一覧を示す図である。

【図6】ページ属性の一覧を示す図である。

【図7】ブックファイルを開く手順例を示すフローチャートである。

【図8】新規のブックファイルを開いた際のユーザインターフェース画面の一例を示す図である。

【図9】既存のブックファイルを開いた際のユーザインターフェース画面の一例を示す図である。

【図10】電子原稿ファイルをブックファイルにインポートする手順例を示すフローチャートである。

【図11】図10のステップ801に示すアプリケーションデータを電子原稿ファイルに変換する手順例を示すフローチャートである。

【図12】印刷や表示を行う際に使用するデータ構造の例を示す図である。

【図13】クライアントサーバ型の文書処理システムのソフトウェア構成例を示すブロック図である。

【図14】データ構成の最上位階層である文書全体の設定を行うダイアログウィンドウの例を示す図である。

【図15】データ構成の中間階層である章の設定を行うダイアログウィンドウの例を示す図である。

【図16】図15のダイアログウィンドウで固有属性の設定を行った例を示す図である。

【図17】データ構成の最下位階層であるページの設定

を行うダイアログウィンドウの例を示す図である。

【図18】図17のダイアログウィンドウでの設定値の変更例を示す図である。

【図19】章やページに固有の設定を行った場合の表示例を示す図である。

【図20A】コピーペーストをされる文書の例を示す図である。

【図20B】固有の設定がないコピーペーストをする文書の例を示す図である。

【図20C】固有の設定がない場合にコピーペーストを行った場合の実行結果の例を示す図である。

【図20D】固有の設定がない場合にコピーペーストを行った場合の実行結果の例を示す図である。

【図21A】コピーペーストをされる文書の例を示す図である。

【図21B】固有の設定があるコピーペーストをする文書の例を示す図である。

【図21C】固有の設定がある場合にコピーペーストを行った場合の実行結果の例を示す図である。

【図21D】固有の設定がある場合にコピーペーストを行った場合の実行結果の例を示す図である。

【図21E】固有の設定がある場合にコピーペーストを行った場合の実行結果の例を示す図である。

【図22】データの挿入処理の例を示すフローチャートである。

【図23】文書全体をペーストする際の処理の例を示すフローチャートである。

【図24】章をペーストする際の処理の例を示すフローチャートである。

【図25】ページをペーストする際の処理の例を示すフローチャートである。

【図26】編集処理における表示方式の例を示す図である。

【図27】編集処理における表示方式の例を示す図である。

【図28】編集処理における表示方式の例を示す図である。

【図29】編集処理における表示方式の例を示す図である。

【図30】編集処理における表示方式の例を示す図である。

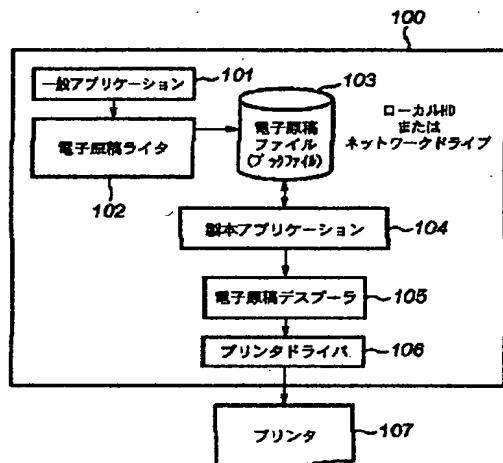
【図31】編集処理における表示方式の例を示す図である。

【図32】編集処理における表示方式の例を示す図である。

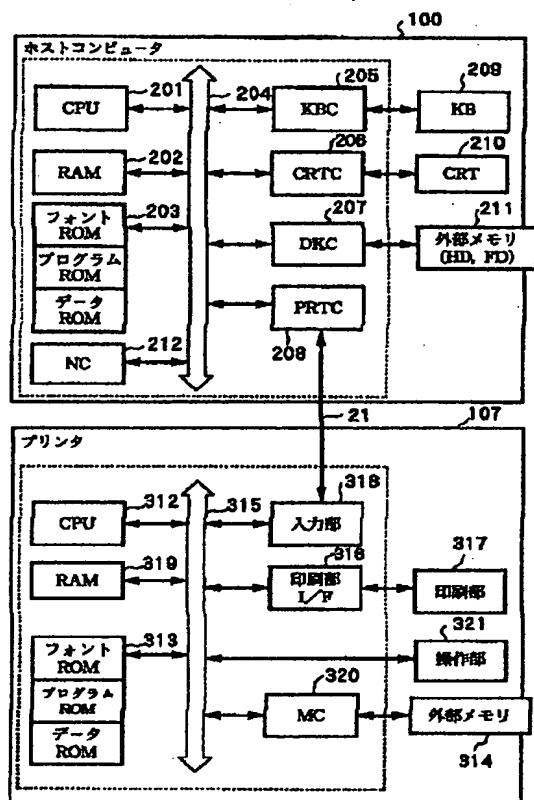
【図33】編集処理における表示方式の例を示す図である。

【図34】編集処理における表示方式の手順例を示すフローチャートである。

【図1】



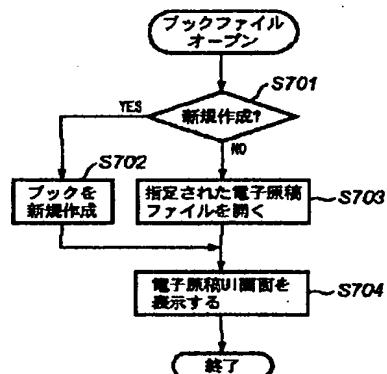
【図2】



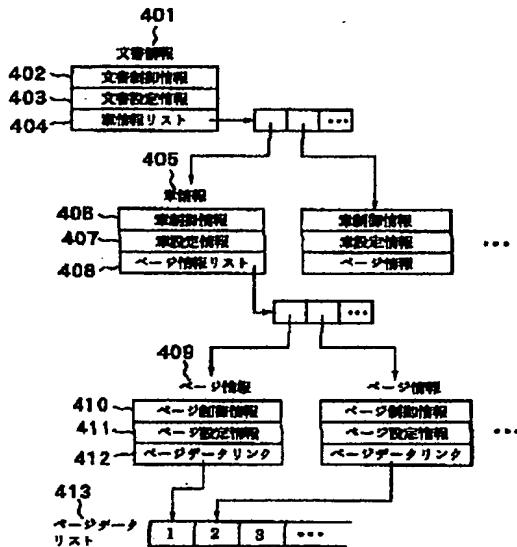
【図4】

No	属性	備考
1	印刷方法	片面/両面/複数印刷
2	用紙サイズ	紙張サイズ/固定サイズ <ul style="list-style-type: none"> 「M4+G」、「B4+G」、「レタートレジャー(11×17)」指定の場合は2折り指定 複数印刷または片面印刷が指定された場合は「1面/1ページ」目的の用紙サイズが自動で選択される
3	用紙方向	縦/横 <ul style="list-style-type: none"> 固定サイズの場合のみ選択可能
4	トリッ代/トリ方向	シフト/逆順指定可能
5	N-up印刷	用紙面積は2/4パターン <ul style="list-style-type: none"> 各奇偶面指定可能
6	紙入端小	ON/OFF <ul style="list-style-type: none"> 用紙サイズに固定サイズのまたはN-up印刷を適用した場合は自動的に指定、OFFに相対可能
7	ウォーターマーク	物理ページ単位、物理ページ単位に個別の指定可能 <ul style="list-style-type: none"> 全面/全ページが対象
8	ヘッダ・フッタ	物理ページ単位、物理ページ単位に個別の指定不可 <ul style="list-style-type: none"> 全面/全ページが対象
9	装録方法	ステイブル/パンチ六 <ul style="list-style-type: none"> ステイブル/パンチは片面/両面印刷のみ ステイブルは片面/両面
10	基本詳細	用紙方向/中とじ/体格指定 <ul style="list-style-type: none"> 複数印刷のみ
11	表紙/裏表紙	表紙1/2、裏表紙1/2に対する印刷指定 <ul style="list-style-type: none"> 表紙口(インサートを含む)指定
12	インデックス紙	インデックス部分への文字列印字、インデックス紙上へのインデックス印字可能 <ul style="list-style-type: none"> 複数印刷不可
13	合紙	紙表口(インサートを含む)指定 <ul style="list-style-type: none"> 複数印刷に静止アーチを印刷可能 複数印刷は指定不可
14	重複切り	「なし」/「ページかえ」/「用紙かえ」 <ul style="list-style-type: none"> インデックス紙、合紙が指定された場合は「用紙かえ」固定 片面印刷は「用紙かえ」

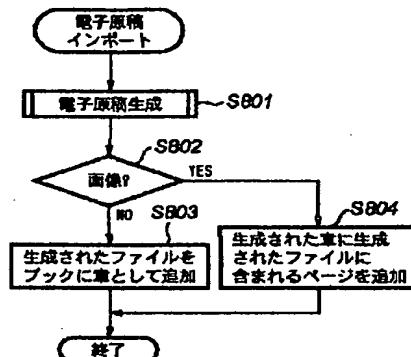
【図7】



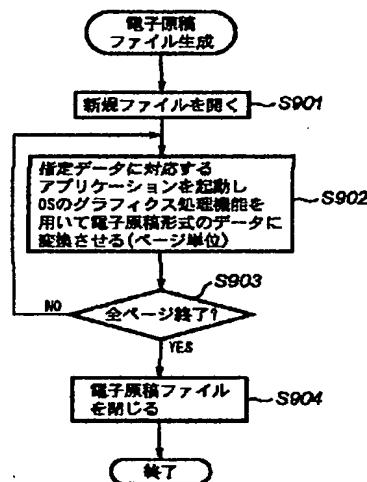
【図3】



【図10】



【図11】



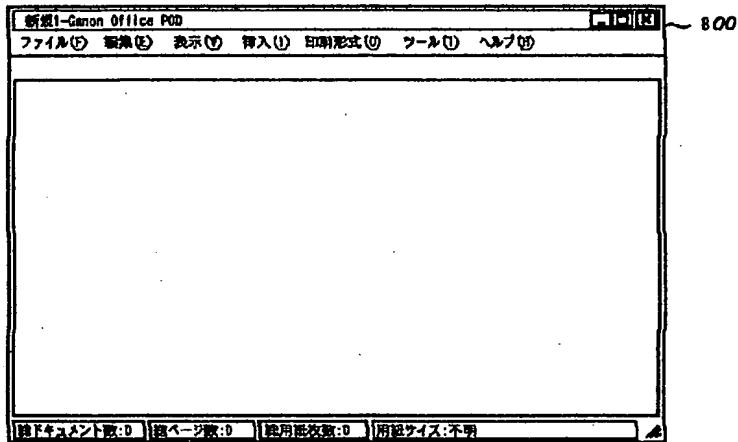
【図5】

No	属性名	備考
1	用紙サイズ	固定サイズ/固定サイズ 固定サイズ選択の場合は「用紙がえ」を自動指定 ブックで複数用紙が選択された場合 指定用紙でのみ変更可能。ブックにあわせる指定 でも用紙サイズの変更可能
2	用紙方向	縦/横 固定サイズの場合のみ選択可能
3	8-up印刷指定	ページ数/記述端/境界線/ 記述位置等 記述位置は3バターン 等割印刷指定可能
4	拡大縮小	DF/DF 用紙サイズに固定サイズのまたはカラー印刷を 選択した場合は自動でDFに指定。DFに指定可能
5	ウォーターマーク	表示/非表示 ブックで指定された全ウォーターマークを表示するか しないかの指定
6	ヘッダー・フッター	表示/非表示 ブックで指定された全ヘッダ・フッタを表示するか しないかの指定
7	接続方法	スタイル ブックでスタイル接続された場合に、DFが可能 デフォルトはON

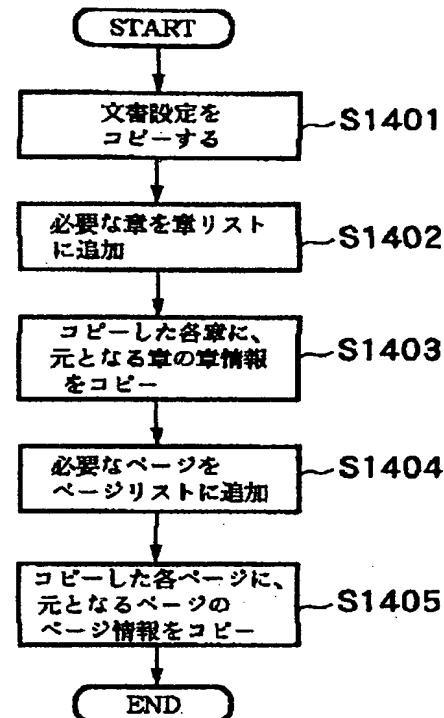
【図6】

No	属性名	備考
1	ページ回転角度	0/90/180/270度の指定が可能
2	ウォーターマーク	表示/非表示 ブックで指定された全ウォーターマークを表示するか しないかの指定
3	ヘッダー・フッター	表示/非表示 ブックで指定された全ヘッダ・フッタを表示するか しないかの指定
4	ズーム	50%-200% 仮想接続ページ領域にフィットしたサイズを 100%とした相对接続指定
5	固定位置	固定バターン右および任意位置指定
6	ノーテーション	
7	Variable項目	
8	ページ分割	

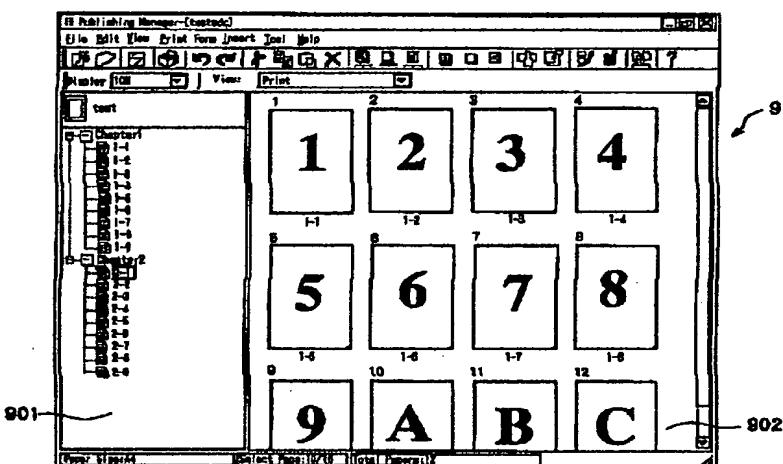
【図8】



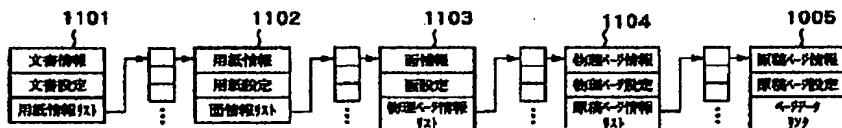
【図22】



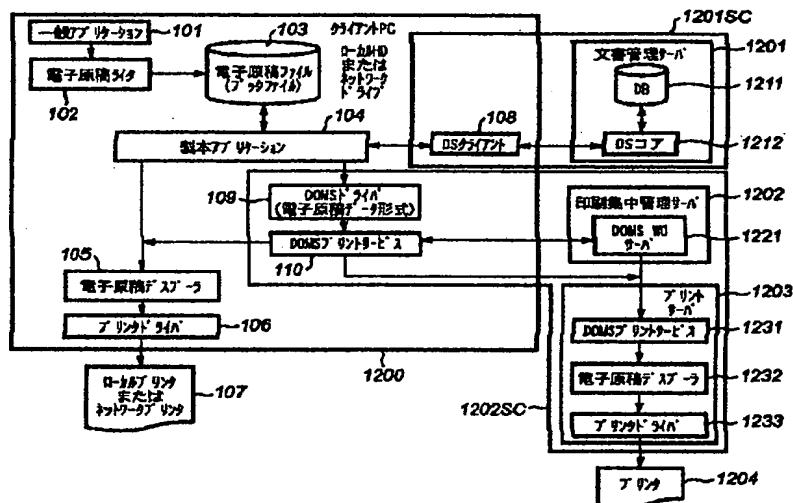
【図9】



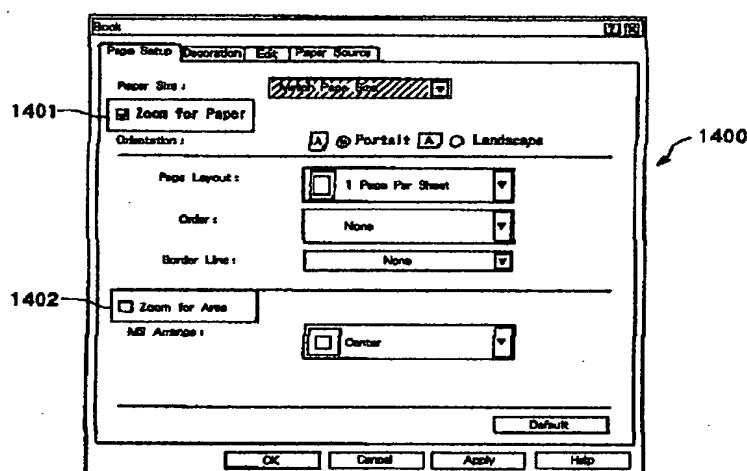
【図12】



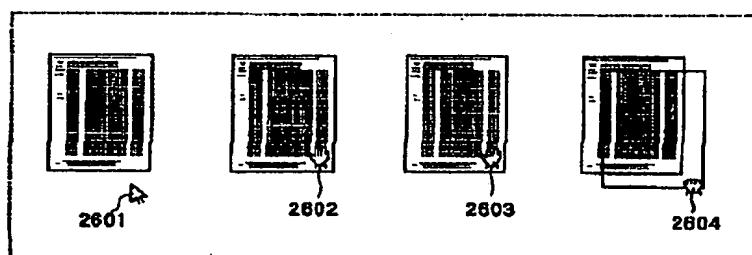
【図13】



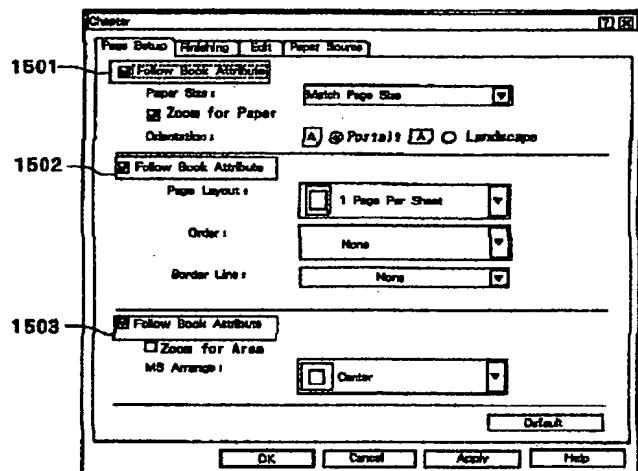
【図14】



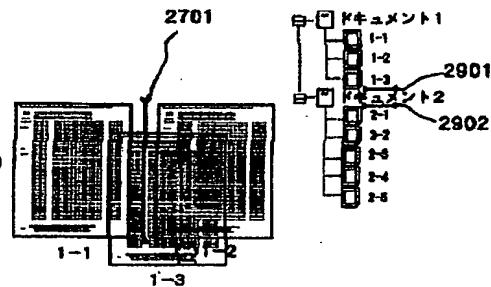
【図26】



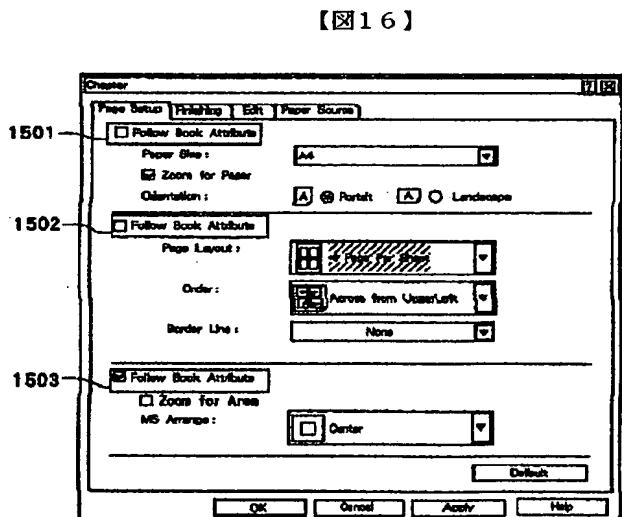
【図15】



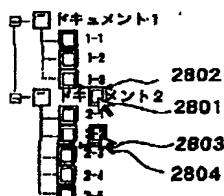
【図27】



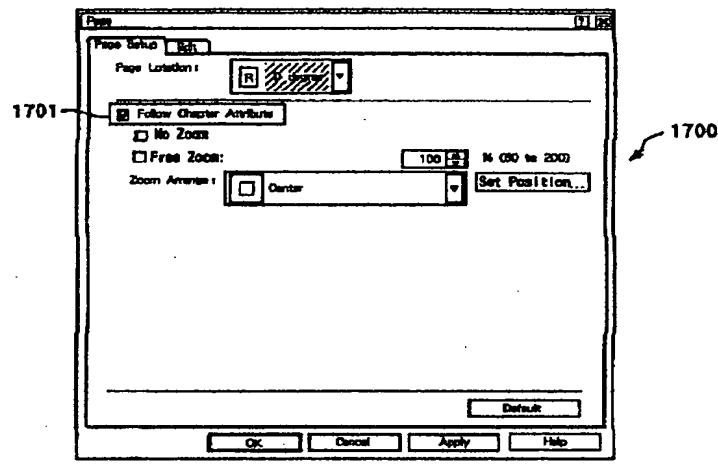
【図29】



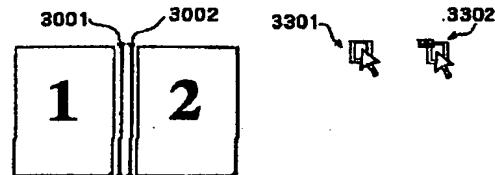
【図28】



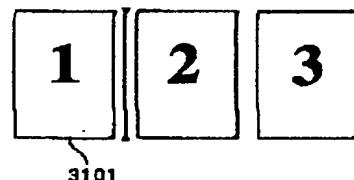
【図17】



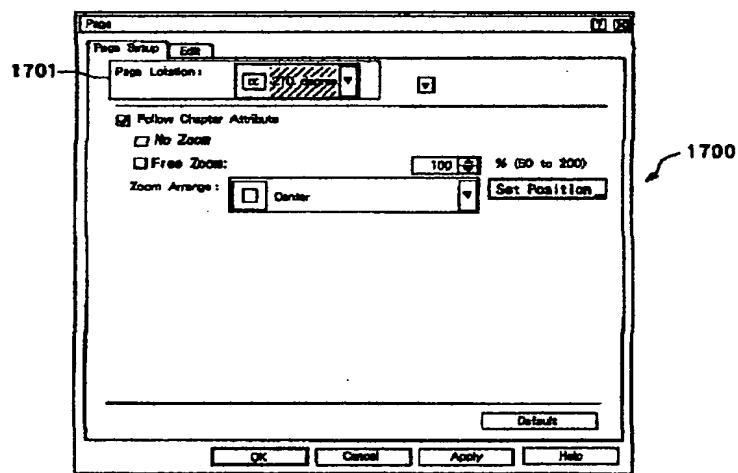
【図30】



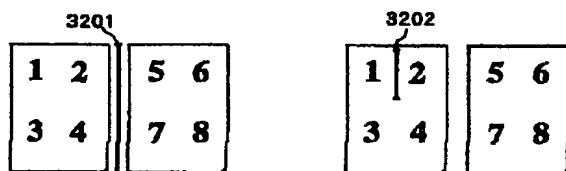
【図33】



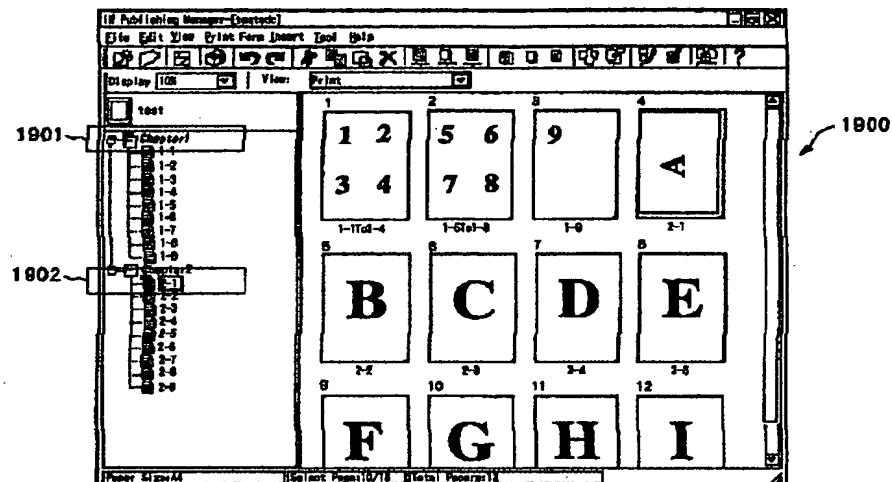
【図18】



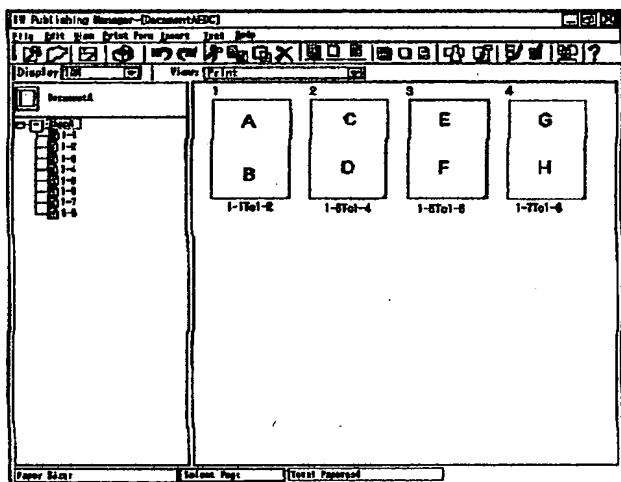
【図32】



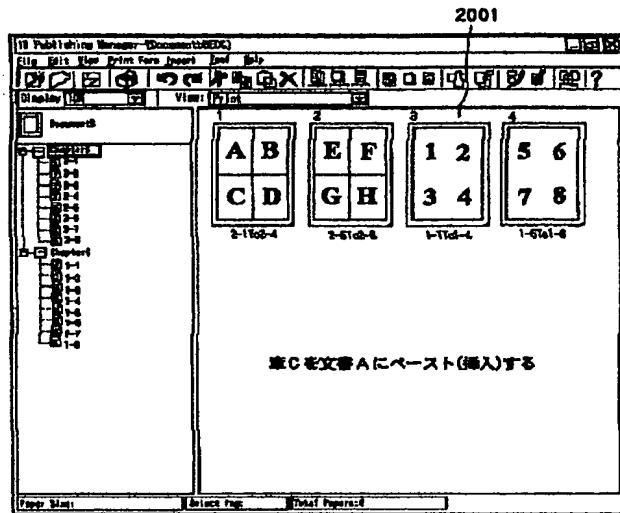
【図19】



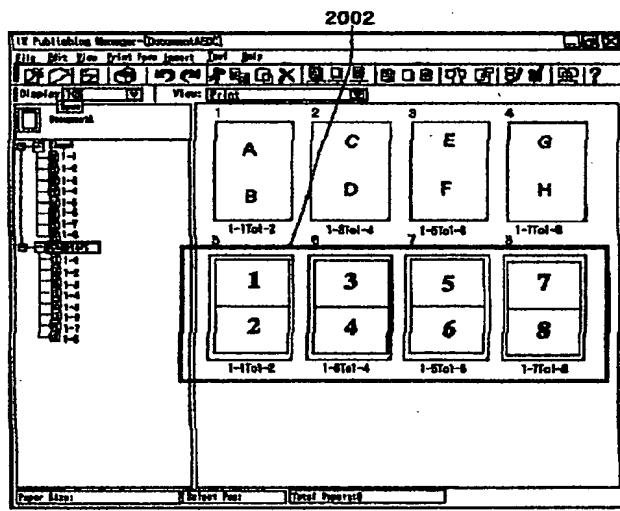
【図20A】



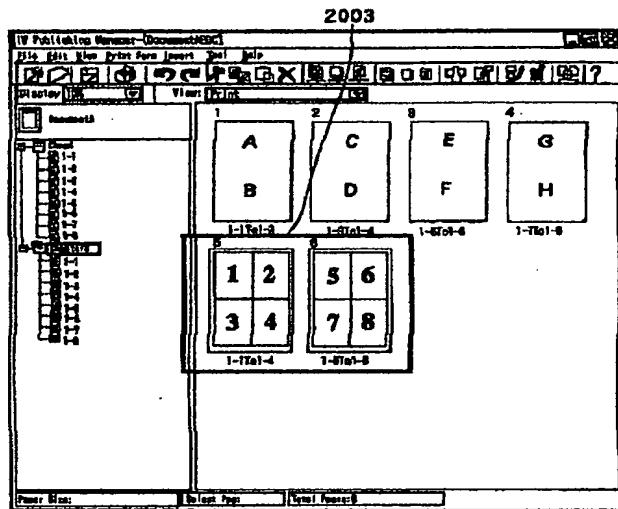
【図20B】



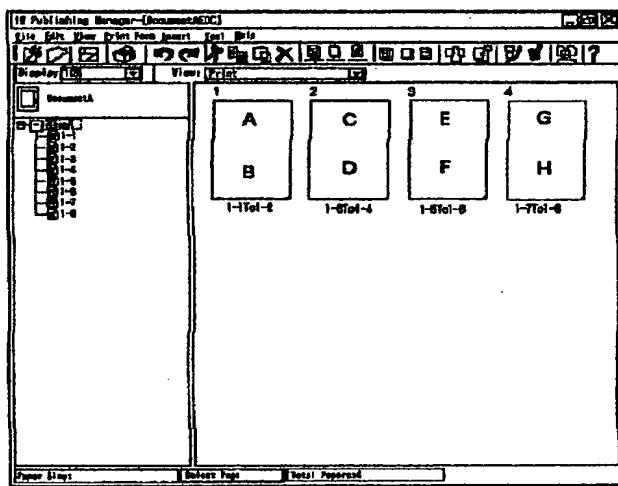
【図20C】



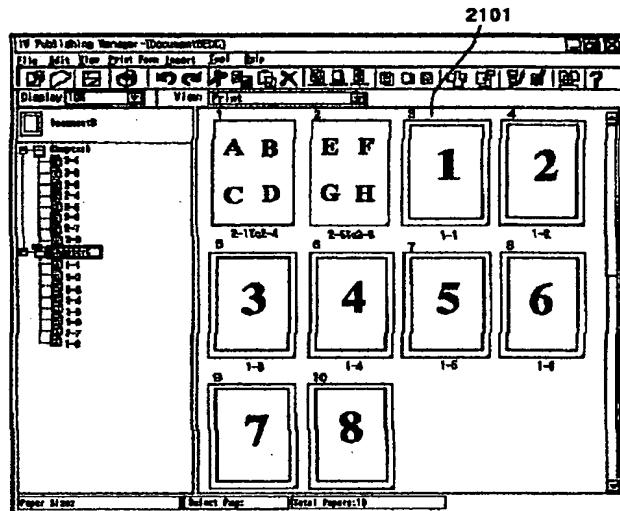
【図20D】



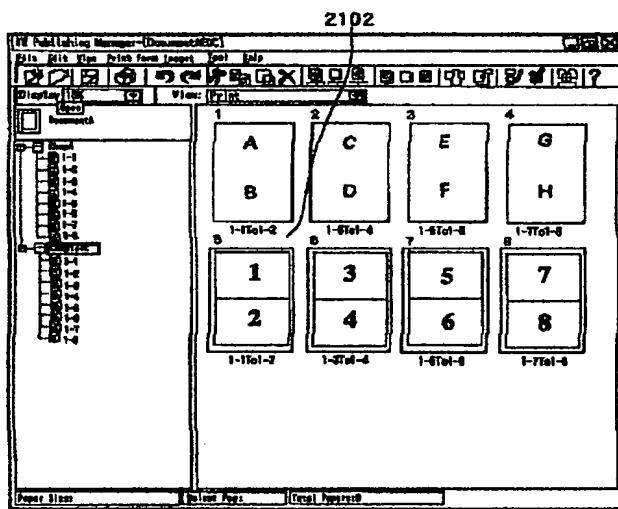
【図21A】



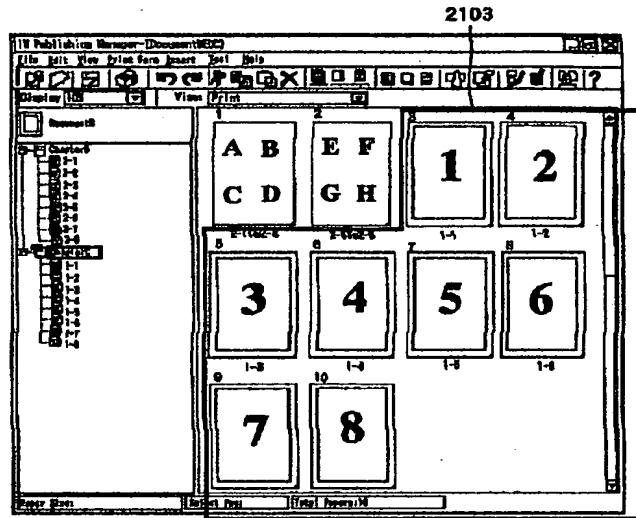
【図21B】



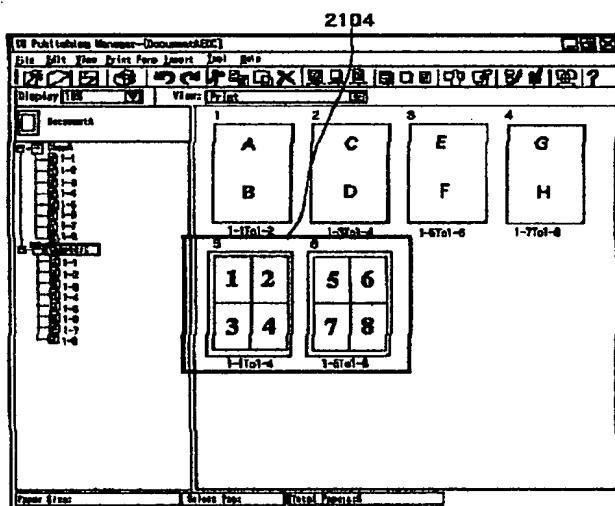
【図21C】



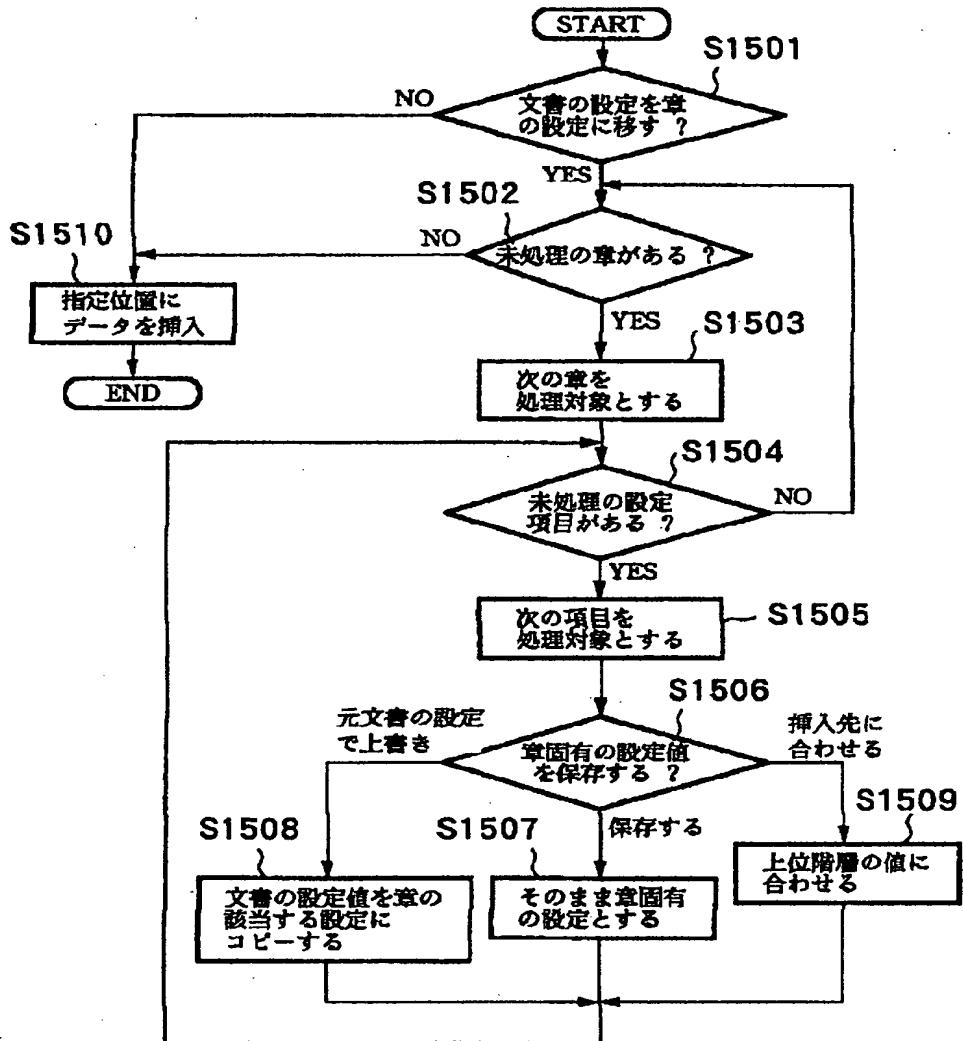
【図21D】



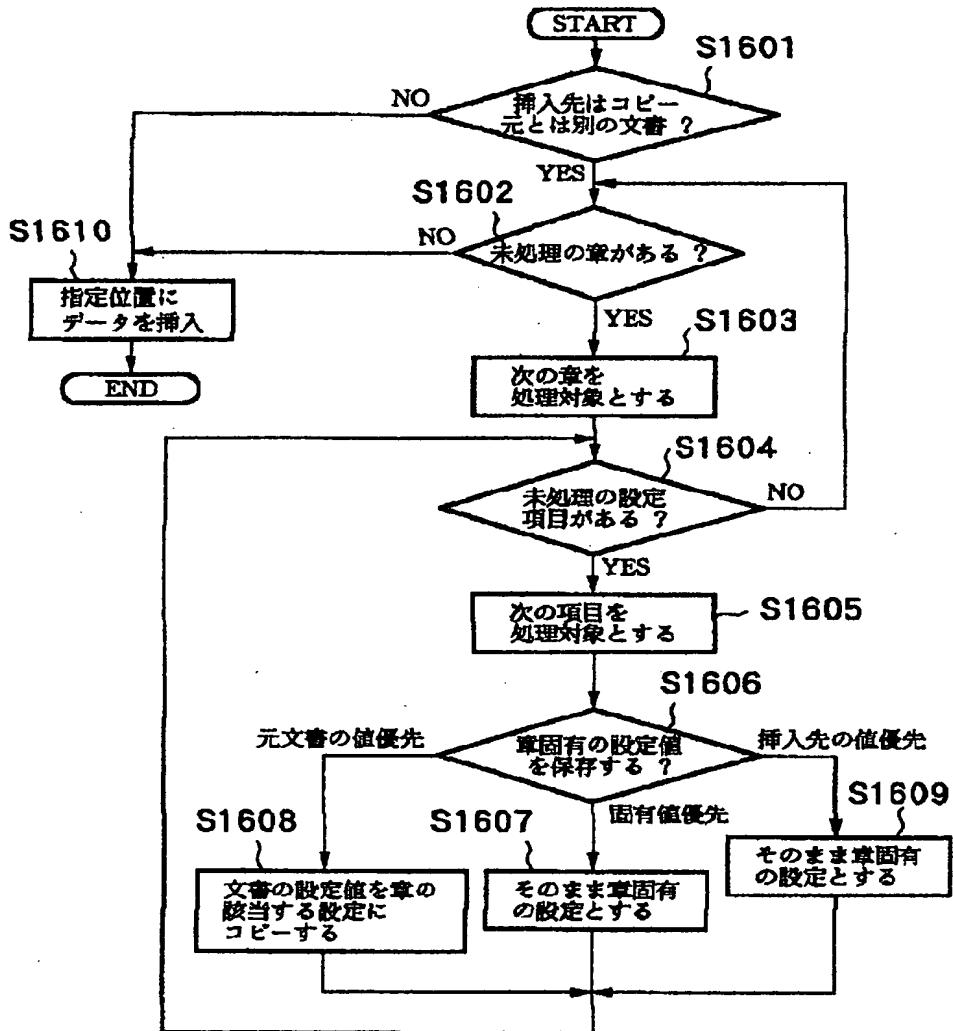
【図21E】



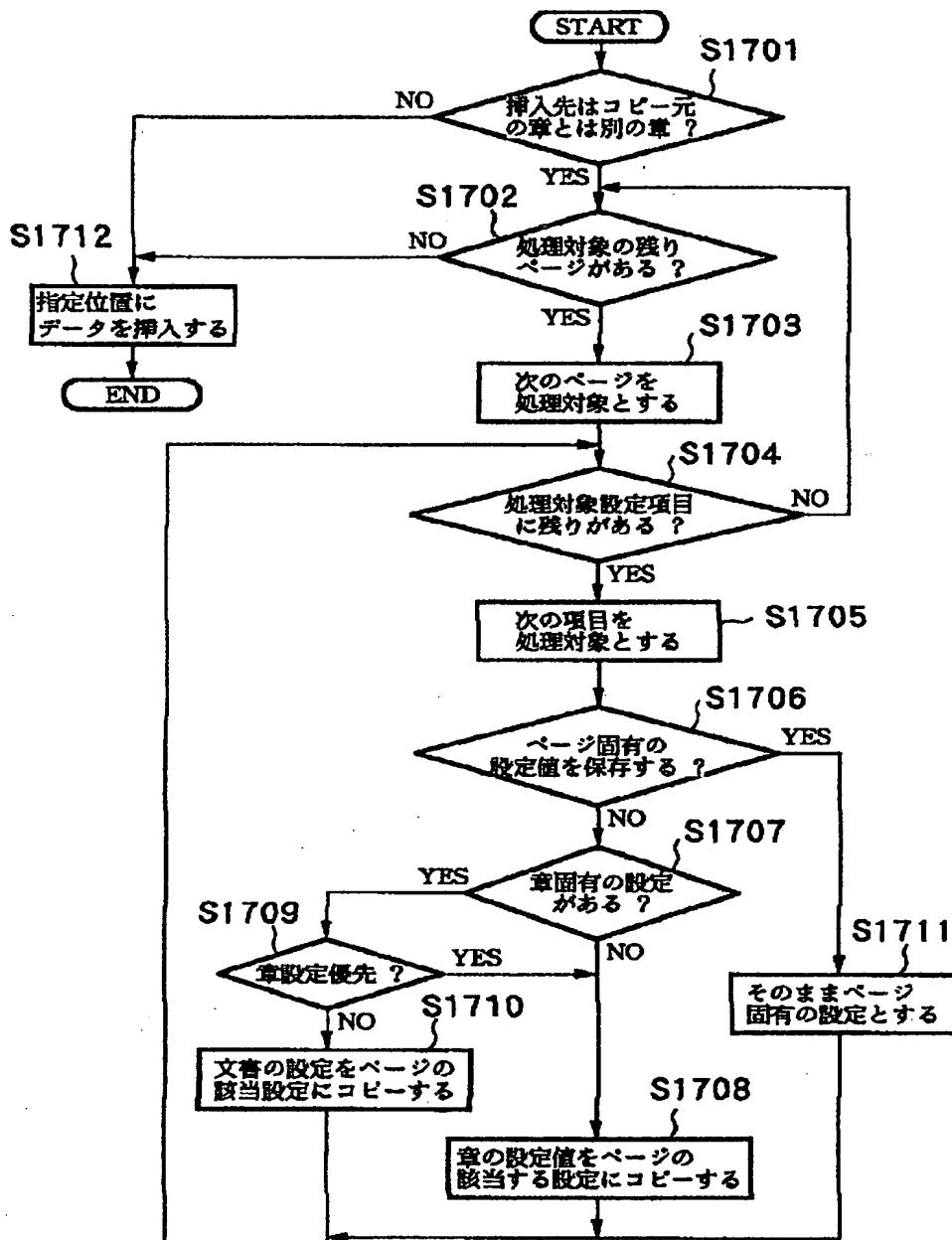
【図23】



【図24】



【図25】



【図34】

